Graduate School Form 9
(Revised 3/11)

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Marc Bosch Ruiz

Entitled

Visual Feature Modeling and Refinement with Application in Dietary Assessment

For the degree of Doctor of Philosophy

Is approved by the final examining committee:

1. Edward J. Delp
Chair
2. David S. Ebert
3. Mary Comer
4. Carol J. Boushey
5.
6.
7.
8.

Format Approved by:

Chair, Final Examining Committee

or Andrew Hughes
Department Thesis Format Advisor

This thesis ☐ **is** ☒ **is not to be regarded as confidential.**

Edward J. Delp
Major Professor

To the best of my knowledge and as understood by the student in the *Research Integrity and Copyright Disclaimer (Graduate School Form 20)*, this thesis/dissertation adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of copyrighted material.

Edward J. Delp
Major Professor

Approved by: Michael R. Melloch 2/15/12
Head of the Graduate Program Date

Graduate School Form 20
(Revised 9/10)

PURDUE UNIVERSITY
GRADUATE SCHOOL

Research Integrity and Copyright Disclaimer

Title of Thesis/Dissertation:
Visual Feature Modeling and Refinement with Application in Dietary Assessment

For the degree of Doctor of Philosophy

I certify that in the preparation of this thesis, I have observed the provisions of *Purdue University Executive Memorandum No. C-22,* September 6, 1991, *Policy on Integrity in Research.**

Further, I certify that this work is free of plagiarism and all materials appearing in this thesis/dissertation have been properly quoted and attributed.

I certify that all copyrighted material incorporated into this thesis/dissertation is in compliance with the United States' copyright law and that I have received written permission from the copyright owners for my use of their work, which is beyond the scope of the law. I agree to indemnify and save harmless Purdue University from any and all claims that may be asserted or that may arise from any copyright violation.

Marc Bosch Ruiz
Printed Name and Signature of Candidate

11/16/2011
Date (month/day/year)

*Located at http://www.purdue.edu/policies/pages/teach_res_outreach/c_22.html

Graduate School ETD Form 9
(Revised 12/07)

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Marc Bosch Ruiz

Entitled
Visual Feature Modeling and Refinement with Application in Dietary Assessment

For the degree of Doctor of Philosophy

Is approved by the final examining committee:

EDWARD J. DELP
Chair

CAROL J. BOUSHEY

DAVID S. EBERT

MARY L. COMER

To the best of my knowledge and as understood by the student in the *Research Integrity and Copyright Disclaimer (Graduate School Form 20)*, this thesis/dissertation adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of copyrighted material.

Approved by Major Professor(s): EDWARD J. DELP

Approved by: M. R. Melloch 02-15-2012
Head of the Graduate Program Date

Graduate School Form 20
(Revised 9/10)

PURDUE UNIVERSITY
GRADUATE SCHOOL

Research Integrity and Copyright Disclaimer

Title of Thesis/Dissertation:
Visual Feature Modeling and Refinement with Application in Dietary Assessment

For the degree of Doctor of Philosophy

I certify that in the preparation of this thesis, I have observed the provisions of *Purdue University Executive Memorandum No. C-22,* September 6, 1991, *Policy on Integrity in Research.**

Further, I certify that this work is free of plagiarism and all materials appearing in this thesis/dissertation have been properly quoted and attributed.

I certify that all copyrighted material incorporated into this thesis/dissertation is in compliance with the United States' copyright law and that I have received written permission from the copyright owners for my use of their work, which is beyond the scope of the law. I agree to indemnify and save harmless Purdue University from any and all claims that may be asserted or that may arise from any copyright violation.

Marc Bosch Ruiz
Printed Name and Signature of Candidate

02-15-2012
Date (month/day/year)

*Located at http://www.purdue.edu/policies/pages/teach_res_outreach/c_22.html

VISUAL FEATURE MODELING AND REFINEMENT WITH APPLICATION IN DIETARY ASSESSMENT

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Marc Bosch Ruiz

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

May 2012

Purdue University

West Lafayette, Indiana

Has d'arribar-hi, és el teu destí,
però no forcis gens la travessia.
És preferible que duri molts anys,
que siguis vell quan fondegis l'illa,
ric de tot el que hauràs guanyat
fent el camí, sense esperar
que et doni més riqueses.
Itaca t'ha donat el bell viatge,
sense ella no hauries sortit.
I si la trobes pobra, no és que Itaca
t'hagi enganyat. Savi, com bé t'has fet,
sabràs el que volen dir les Itaques.

-Camí d'Itaca-

To my parents, sister
and wife, Kara.

ACKNOWLEDGMENTS

My future endeavors as an engineer will be greatly influenced by all of what my advisor has taught me. There are many things that I will not accomplish in a lifetime, but I can proudly say that I was Professor Edward J. Delp's student. I am grateful for his support and guidance. He has helped me grow incredibly during my time here and the advice he provided has been valued and appreciated beyond words. With him I have learned a new way of thinking: there are very few things in this world that they cannot be accomplished. Professor Delp, thank you very much.

I would also like to thank Professor Carol J. Boushey. Working with her has given me the opportunity to learn how important the details are in order to develop great tools. I have also learned how to incorporate the "user component" in our engineering world. But the most important thing I have learned from her is how to work in a multidisciplinary environment and how sharing ideas from different fields can help us obtain new perspectives in a research project.

I am grateful to my advisory committee members: Professor Mary L. Comer, and Professor David S. Ebert for their valuable suggestions, questions, and support. I would like to thank Purdue University, the Graduate School, and the School of Electrical and Computer Engineering for accepting me into the Masters and Doctoral programs. I am thankful to Professor Lluís Torres for encouraging me to go to Purdue University, definitely one of the best pieces of advice I have ever received.

It has been a pleasure working with my truly great colleagues at the VIPER laboratory. Special thanks to Dr. Nitin Khanna for working with me on many papers, for giving always very constructive feedback in every research step, and for being on top of things. Thanks to Dr. Fengqing Maggie Zhu, my long term co-worker, with whom I have been through all stages of our research project. Also I would like to thank Kevin Lorenz and Aravind Mikkilineni for having an answer to all my IT

troubles. I also want to thank Albert Parra my fellow Catalan in the lab, colleague, roommate, and friend. And thanks to the rest of former and current colleagues: Dr. Golnaz Abdollahian, Ziad Ahmad, Dr. Ying Chen Lou, Andrew Haddad, Ye He, Deen King-Smith, Liang Liang, Limin Liu, Dr. Anthony Martone, Anand Mariappan, Dr. Ka Ki Ng, Dr. Satyam Srivastava, Carlos Wang, Meilin Yang, Chang Xu, and Bin Zhao. Thanks to all the visitors in the lab that I have interacted with: Murat Birinci, Professor Fernando Díaz de María, Professor Moncef Gabbouj, Professor Josep Prades Nebot, Antoni Roca, and Francisco Serrano.

I would like to extend my gratitude to the entire TADA team: Elisa Bastian, Ashley Chambers, Junghoon Chae, Dr. Heather Eicher-Miller, Shivangi Kelkar, SungYe Kim, Professor Martin Okos, Bethany Six, Scott Stella, Karl Ostmo and Insoo Woo. Special thanks to TusaRebecca Schap for her great ideas on developing the TADA project, and all those shared moments during our demos. Thank you to our great partners in Australia: Professor Deborah Kerr, Katherine Kerr, Greg Kerr, and Professor Mark Pickering.

I would like to thank the National Institutes of Health (NIH) for funding the project which resulted in this thesis.

I thank all my friends, from here and there, years-long and recent, for their affection, friendship, and for making me realize of the 'big picture' of many things in the good times and in the not-so-good times.

I would like to thank my parents for ... everything. In particular, for encouraging me to do what I wanted to do wherever that was. They have given me more than I could ever imagine and they have taught me how to be a good person. I would like to thank my sister, Claudia, for being such a good person. I am grateful to all my family in Spain (Bosch Ruiz) and in the USA (Cunzeman) for their perpetual love and support. Thanks to everyone who has, at some point of time, helped me grow up. I would like to also express gratitude to the people of the United States for welcoming to their country.

Last, but by no means least, my wife, Kara Cunzeman, There are literally no words to express my gratitude for being always there, and for her friendship and love. I honestly believe that without her this would have been a much, much more difficult road. I would like to thank her for bringing happiness into my life and for taking care of me at all times.

This work was sponsored by grants from the National Institutes of Health under grants NIDDK 1R01DK073711-01A1 and NCI 1U01CA130784-01.

TABLE OF CONTENTS

Page

LIST OF TABLES x

LIST OF FIGURES xii

ABBREVIATIONS xvii

ABSTRACT xix

1 INTRODUCTION 1

1.1 Overview 1

1.1.1 Traditional Methods for Dietary Assessment 1

1.1.2 Technology Based Methods For Dietary Assessment 2

1.1.3 Image Analysis For Food Recognition 3

1.2 Object Recognition 6

1.2.1 Feature Description for Object Characterization 6

1.2.2 Object Classification 9

1.2.3 Feature Extraction and Classification Model 11

1.2.4 Multiple Hypothesis Segmentation and Classification 12

1.3 The TADA System 14

1.3.1 TADA System Architecture 15

1.3.2 Integrated Food Image Databases 17

1.4 Contributions Of This Thesis 18

1.5 Publications Resulting From This Work 19

2 FEATURES FOR OBJECT CHARACTERIZATION 22

2.1 Features for Visual Characterization 22

2.2 Global Features 23

2.2.1 Color Features 24

2.2.2 Texture Features 30

Page

2.3 Local Features 46
2.3.1 Feature Detection and Points of Interest 48
2.3.2 Local Descriptors 53
3 OBJECT CLASSIFICATION 66
3.1 Background 66
3.2 Individual Feature Channel Classification 67
3.2.1 k-Nearest Neighbor (KNN) 69
3.2.2 Support Vector Machine (SVM) 70
3.2.3 Bag-of-Features (BoF) 72
3.3 Combination Of Local And Global Features 79
3.3.1 Classifier Confidence Measure 80
3.3.2 Late Decision Fusion 81
3.4 Context-Based Refinement 81
4 EXPERIMENTAL RESULTS 87
4.1 Color Features Evaluation 87
4.1.1 Datasets Used For Evaluation Of The Color Features 87
4.1.2 Color Feature Performance 88
4.2 Texture Analysis 91
4.2.1 Datasets For Evaluation Of The Texture Features 92
4.2.2 Texture Descriptors Evaluation 94
4.3 Local Descriptors and the Bag-of-Features (BoF) Evaluation 97
4.3.1 Point Detector Evaluation 97
4.3.2 Local Descriptor Evaluation 100
4.4 Overall Object Classification 103
4.4.1 Datasets For Multi-Channel Classification Evaluation 103
4.4.2 Performance Evaluation Of Each Individual Channel 107
4.4.3 Decision Fusion Evaluation 109
4.4.4 Codebook Refinement 113

Page

4.4.5 System Performance With Contextual Information 113
4.5 System Evaluation: User Study 114
4.5.1 Classification Results: Controlled User Study 115
4.5.2 Classification Results : Free-living User Study 115
5 INTEGRATED IMAGE DATABASES FOR DIETARY ASSESSMENT . 122
5.1 TADA Databases 122
5.1.1 I-TADA 124
5.1.2 T-FNDDS 127
5.1.3 E-TADA 131
5.2 TADA System Architecture 134
5.2.1 Mobile User Interface 134
5.2.2 TADA System Integration 136
5.2.3 The TADA Databases: Current Deployment 142
5.2.4 User Study Validation 143
6 CONCLUSIONS AND FUTURE WORK 147
6.1 Conclusions 147
6.2 Future Work 149
6.3 Publications Resulting From This Work 152
LIST OF REFERENCES 156
A DATABASE SOFTWARE 169
VITA 170

LIST OF TABLES

Table Page

2.1 List of feature channels investigated and their type (global color, global texture or local). 65

4.1 Average classification rate of color features using 1-Nearest Neighbor with L_2 norm. There are 37 food categories. Experiments were performed with random selection of training and testing data, as well as increasing the percentage of training data used (25%, 50% and 75%). 90

4.2 Classification rates using GLCM, Gabor, MFS, EFD, GFD, and GOSDM for food dataset, Brodatz, and UIUC using different training data percentages (40%, and 80%). Feature vector dimension is also included. 96

4.3 Average classification rate of the *DoG* and *entropy points* detectors with *Steerable filters* and *SIFT* descriptors using BoF with KNN for classification. Experiments were performed with random selection of training and testing data, as well as increasing the percentage of training data (25%, 50%, and 75%). 98

4.4 Average classification rate of all local features using BoF model for classification and *DoG* as point detector. Two signatures are compared and classified using KNN. Experiments were performed with random selection of training data. 101

4.5 Queries used to retrieve images from Flickr. Words in bold show the "targeted" class 4.5. 106

4.6 Mean classification rate for all classes for each type of feature channel using a KNN and SVM classifiers. L indicates local feature. G indicates global feature. 108

4.7 Average classification rate for each decision fusion approach Majority vote rule and Maximum confidence score for both KNN and SVM classifiers for multiple candidates (1, and 8) 109

4.8 Percentage of agreement of each feature channel classifier with the final decision after fusion using 1, 4, and 8 candidates. Note these are percentage averaged for all classes in the food database 112

4.9 Identification Rate using the top 1, and 4 class suggestions given by the classifier. 116

Table Page

5.1 Pin manipulation actions of the top 4 users, bottom 4 users and average of all users in the Spring/Summer 2011 free living study. **Added** refers to pins added by the user due to mis detection of food items in the segmentation step, or foods not in our database, **Removed** pins removed by the user because they are pointing to a non food item, **Confirmed** refers to all pins that were correct, **Suggested** pins that were changed but the correct food was within the suggested foods, and finally, **Changed** refers to pins that were changed and not in the suggested foods list. 145

LIST OF FIGURES

Figure Page

1.1 Multiple Hypothesis Segmentation and Classification System. 13

1.2 The TADA System Architecture. 16

2.1 Two Samples Of The Same Food (*Vegetable Soup*) With Different Color Distributions. 29

2.2 Examples Of Food Items Represented Using Predominant Colors. (a,b) Spaghetti. (c,d) Pears. (e,f) Scrambled Eggs. 31

2.3 Two Samples Of The Same Texture Only Differing By Their Gray Level. 33

2.4 GOSDM Block Diagram. 34

2.5 Three Examples Of Gradient Orientation Fields. (a) Corn Texture, (b) Watermelon Texture, And (c) Spaghetti Texture. Note That Color Information Has Been Included For Illustration Purposes. For The Gradient Estimation The Image Is First Transformed To Gray Scale Space. . . 35

2.6 Dominant Orientation Estimation Method To Achieve Robustness Against Rotation. 36

2.7 Examples Of GOSDMs For Various Textures. a) Original Textures. b) Corresponding GOSDM With $d = (4, 0)$. (All images are converted to gray scale in our classification scheme). 37

2.8 Comparison Of Four Food Textures At Two Scale Levels. (a) Small Scale Cauliflower, (b) Large Scale Cauliflower, (c) Small Scale Popcorn, (d) Large Scale Popcorn, (e) Small Scale Mayo, (f) Large Scale Mayo, (g) Small Scale String Cheese, (h) Large Scale String Cheese. 38

2.9 Entropy-Based Multifractal Analysis Block Diagram (EFD). 43

2.10 Gabor-Based Multifractal Analysis Block Diagram (GFD). 46

2.11 Examples Of Gabor Features. (a) Original Textures. (b) Mean Of The Energy. (c) Standard Deviation Of The Energy. (d) GFD Signature. . 47

2.12 Example Of Gaussian Filter. 49

2.13 Difference Of Gaussian Formation [138]. 51

2.14 Example Of Difference Of Gaussians To Locate Points Of Interest. . . 52

Figure Page

2.15 Entropy-Based Point Detector. 54

2.16 SIFT Descriptor Representation [138]. (Left) Gradient Orientations And Magnitudes Are Estimated At Each Pixel Location And Weighted By A Gaussian Fall-off Function (Blue Circle). (Right) A Weighted Gradient Orientation Histogram Is Then Computed In Each Subneighborhood Using Interpolation. Note That This Figure Shows An 8×8 Pixel Patch And A 2×2 Descriptor Array, Lowe In His Experiments Used A 16×16 Pixel Patch And A 4×4 Array Of Eight Bin Histogram. 56

2.17 Haar Filters. (a) Horizontal. (b) Vertical. 56

2.18 The DAISY Descriptor. Each Circle Represents A Region Where The Radius Is Proportional To The Standard Deviations Of The Gaussian Kernels. The + Sign Represents The Locations Where The Convolved Orientation Maps Center Are Sampled. Around These Locations The Descriptor Is Estimated [146]. 62

2.19 Filter Definitions. (a) Neighborhood Division. (b) Filters Coefficients Used. 64

3.1 Food Classification System. (LG is the number of global feature channels, and LL is the number of local feature channels. $f'(.)$ corresponds to the training feature set, and $f(.)$ corresponds to features of the image.) . . 68

3.2 Bag Of Visual Words Obtained From Local Features. 73

3.3 Bag Of Features Model. 73

3.4 3 Full Hierarchical k-means Propagations [160]. 74

3.5 Example Of The Visual Word Hierarchical Tree For A Database Containing 452 Objects, 1110 Visual Words, And Using SIFT As The Feature Space. 75

3.6 Contextual Information Examples: a) Object Combinations, b) Object Misclassifications (right). 85

4.1 Examples Of 37 Food Classes Used In The Color Experiments.(From left to right and top to bottom: *Apple Juice, Bagel, BBQ Chicken, Broccoli, Brownie, Canned Pear, Catalina Dressing, Chocolate Cake, Coke, Cream Cheese, Egg, French Dressing, French Fries, Fruit Cocktail, Garlic Bread, Gravy Chicken, Green Beans, Hamburger, Ketchup, Lettuce, Mac and Cheese, Margarine, Mashed Potatoes, Milk, Orange Juice, Peach, Peanut Butter, Pineapple, Pork Chop, Regular Coffee, Sausage, Spaghetti, Strawberry Jam, Sugar Cookie, Toast, Vegetable Soup, Yellow Cake*). . 89

4.2 Examples Of Our Customized Food Texture Dataset. 93

Figure Page

4.3 Examples Of The Brodatz Dataset Texture Samples. 94

4.4 Examples Of The UIUC Dataset Texture Samples. 94

4.5 Example Of Point Detector Density For All The Images In The Dataset: *DoG* (left), And *Entropy* Points (right). Point Detector Density Is The Ratio Between Number Of Points Detected And Size In Pixels Of The Segmented Region. 99

4.6 Example of point detectors: *DoG* (left), and *Entropy* points (right). . 100

4.7 Examples Of 83 Food Classes. (From left to right and top to bottom: *Apple, Apple juice, Bagel, Banana, BBQ Chicken, Broccoli, Brownie, Carrot, Celery, Cheese Burger, Chicken Wrap, Chocolate Cake, Chocolate Chip Cookie, Clementine, Coffee, Coke, Cream Cheese, Cup, Egg (Scrambled), English Muffin, French Dressing, French Fries, Frozen Meat Loaf, Frozen Meal Turkey, Fruit Cocktail, Garlic Bread, Glass, Goldfish, Granola Bar, Grapes, Gravy Chicken, Green Beans, Ham Sandwich, Ice Cream, Jelly, Ketchup, Lasagna, Lettuce, Mac and Cheese, Margarine, Mashed Potatoes, Mayo, Milk, Muffin, Non Fat Dressing, Noodle Soup, Orange, Orange Juice, Pancake, Peach, Peanut Butter, Pear, Peas, Pineapple, Pizza, Plate, Pork Chop, Potato Chips, Pretzel, Pudding, Ranch Dressing, Rice Krispy Bar, Salad Mix, Saltines, Sausages, Snicker Doodle, Snicker, Spaghetti, Strawberry, Strawberry Jam, String Cheese, Sugar Cookie, Syrup, Utensil, Vegetable Soup, Watermelon, Wheat Bread, Wheaties, White Toast, Yellow Cake, and Yogurt.* 104

4.8 Examples Of The 20 Categories In The PASCAL Database. (From left to right and top to bottom: *Aeroplane, Bicycle, Bird, Boat, Bottle, Bus, Car, Cat, Chair, Cow, Dog, Horse, Motorbike, Person, Sheep, Sofa, Table, Potted Plant, Train*, and *TV/monitor*) 105

4.9 The Effect Over $k = 5$ Nearest Neighbors By (a) Max Confidence Score (Real configuration) And (b) Majority Vote Rule Criteria. 110

4.10 Examples Of Misclassified Food Segmented Regions. 112

4.11 Signature Distances For SIFT Features Using (a) Hierarchical K-means And (b) KECA Clustering. 117

4.12 An Example Of Misclassified Segmented Regions After Context Refinement (Left Image), And Corrected Misclassifications By Contextual Information (Right Image). Red Labels Represent Misclassifications And Green Labels Represent Correctly Labeled Segmented Regions. 118

4.13 Examples of Meal Images In A Controlled User Study. 118

Figure Page

4.14 Confusion Matrix For Controlled User Studies. 28 Food Classes. Groundtruth Segmentation. 119

4.15 Examples of Eating Occasion Images Acquired In Free-living User Study. 119

4.16 Identification Rate for Each Participant using (a) Top 1 and (b) Top 4 Food Classes as Classifier Output. 120

4.17 Examples Of Difficult Images To Classify Due To Different Lighting Conditions (1, 2, 3 , 4, and 6), Cluttered Background (1, 4), Blurred Images (3, 5), And Different Food Setups - Colored plates Or No Plates At All - (7, 8, and 9). 121

5.1 The Main Components And Contents For Each TADA Database. . . . 123

5.2 An Example Of The Information Available In I-TADA For A Particular Image. 125

5.3 Examples of EXIF data available for three digital cameras. 126

5.4 Example Of Some Of The Nutrition Information Available In The T-FNDDS. 128

5.5 Example VCM Integrated Into FNDDS To Provide Both Nutritional And Visual Description Of The Food Items. Using the XML Language Can Be Incorporated Into A Web-based Application Available To Researchers. 129

5.6 Illustration of the Visual Characterization Metric (VCM) and the Structure for Visual Descriptions for a Food Image. 130

5.7 An Example of a VCM Integrated into FNDDS to Provide Both Nutritional and Visual Description of the Food Items. Using XML the VCM Can Be Incorporated into a Web-based Application. 131

5.8 Barcode Information Integrated Into The T-FNDDS. 132

5.9 Examples of Data Available for Each User in *E-TADA* for the Free-living Studies. 133

5.10 Examples of Data Available for a Controlled Studies Including Menus Served. 134

5.11 Examples Of The Mobile User Interface For Dietary Assessment (mdFR). (a) Home Screen - Record Eating Events Mode, (b) List View Of All Unconfirmed Eating Occasions - Review Mode (c) Review Results- Review Mode. 137

5.12 Databases interaction with the TADA system. 138

5.13 Examples of User Specific Statistics Available via the Web Interface. . 141

Figure Page

5.14 GPS Location Connected To An Online Map Service For Visualization. 142

5.15 Example Of Email Notifications Sent To The Researcher With Participant's Information. 143

5.16 An Example Of Classification Refinement Using User's Feedback. (a) Original Image, (b) Mask After Image Analysis, (c) User Feedback (Mask And Local Labels), And (d) Mask After Classification Refinement. . . 146

6.1 Block Diagram Of A Personalized Classifier Using Contextual Information. 150

6.2 Examples Of Training Images Retrieved With Google Search (a) *Brownie*, and (b) *Lasagna*. 151

ABBREVIATIONS

BoF	Bag of Features
CRF	Conditional Random Fields
DDL	Description Definition Languages
DMB	Database Management System
DoG	Difference of Gaussians
EFD	Entropy-based categorization and Fractal Dimension (EFD)
E-TADA	Experiments TADA Database
EXIF	Exchangeable Image File Format
FD	Fractal Dimension
FNDDS	USDA Food and Nutrient Database for Dietary Studies
GFD	Gabor-based image de- composition and Fractal Dimension
GLCM	Gray Level Co-occurrence Matrix
GOSDM	Gradient Orientation Spatial-Dependence Matrix
I-TADA	Image TADA Database
KECA	Kernel Entropy Component Analysis
KNN	k-Nearest Neighbors
LoG	Laplacian of Gaussian
mdFR	Mobile Device Food Record
MFS	Mulifractal Spectrum
MPM	Maximum Posterior Marginal
PCA	Principal Component Analysis
PDA	Personal Digital Assistants
SIFT	Scale Invariant Feature Transform
SURF	Speeded-Up Robust Features

SMS	Short Message Service
SVM	Support Vector Machine
TADA	Technology Assisted Dietary Assessment
T-FNDDS	TADA FNDDS
VCM	Visual Characterization Metric
VOC	Visual Object Classes
XML	Extensible Markup Language

ABSTRACT

Bosch Ruiz, Marc. Ph.D., Purdue University, May 2012. Visual Feature Modeling and Refinement with Application in Dietary Assessment. Major Professor: Edward J. Delp.

There has been rapid emergence of technologies for improving our lives and health. However, the technologies for real-time monitoring of diet are still in their infancy. In this thesis we describe an imaging based tool to assess diet. Meal images taken before and after eating allow for the automatic estimation of consumed foods using image processing and analysis methods. In this thesis we have investigated features for efficient visual characterization of food items, including color, texture and local descriptors. Emphasis is given to textural features by describing three unique texture descriptors for both texture classification and retrieval that can be used to characterize food items. We describe a classification system for food identification that can be extended to other object classification tasks. Multiple feature spaces are independently classified and corresponding decisions are fused together according to a set of rules to achieve a final decision. Potential misclassifications are corrected by using contextual information, such as object interaction, and information from the confusion matrix on the validation dataset. We evaluated our models based on food datasets from controlled and natural eating events and on publicly available object recognition benchmark datasets.

A database architecture is described for capturing and indexing information from our food imaging system. This database system has been complemented with a web interface that allows researchers to monitor patients in real-time, and interact with the dietary data in unique ways. This system provide tools for nutritionists and the

health research community that can be used for further data mining to extract diet pattern of individuals and/or social groups.

1. INTRODUCTION

There is a health crisis in the US related to diet that is further exacerbated by our aging population and sedentary lifestyles. Six of the ten leading causes of death in the United States, including cancer, diabetes, and heart disease, can be directly linked to diet [1, 2]. One way to address this problem is to encourage people to maintain good health and quality of life by participating in the management of long-term health choices rather than the traditional approach of medicating acute conditions with powerful drugs. Dietary intake, the process of determining what someone eats during the course of a day, provides valuable insights for mounting intervention programs for prevention of many of the above chronic diseases. Measuring accurate dietary intake is considered to be an open research problem in the nutrition and health fields. Traditional dietary assessment is comprised of written and orally reported methods that are time consuming and tedious, often requiring a nutrition professional to complete, and are not widely acceptable or feasible for everyday monitoring [3].

1.1 Overview

1.1.1 Traditional Methods for Dietary Assessment

Since 1960 the percentage of overweight and obese adults has increased in the United States from 43.3% in 1960 to 66.3% in 2003-04 [4]. Obesity increases the risk of cardiovascular disease and overall mortality [5]. In particular among the youth the increasing prevalence of obesity is of great concern [6]. A weight loss as modest as 5% can significantly improve the risk for several chronic diseases, such as hypertension, diabetes mellitus and insulin resistance [7,8]. Dietary intake provides valuable insights for mounting intervention programs for prevention of disease.

Dietary records are often use to aid dietary assessment. They are generally considered more accurate than dietary recall for determining energy intake because weighing and measuring food reduces the errors associated with estimating portion size and because the participant records intake immediately after eating, which decreases the number of errors associated with recall [9, 10]. Reporting error does not occur systematically across different age groups or different dietary survey techniques.

Despite technology that has enabled the creation of countless tools and devices that improve our lives and health every day, methods for real-time assessment and proactive health management of diet do not currently exist. Preliminary studies among adolescents suggest that innovative use of technology may improve the accuracy of diet information from young people [11].

1.1.2 Technology Based Methods For Dietary Assessment

As mentioned in the previous section assessment of diet is problematic and is still an open research problem. The use of modern technology, such as mobile telephone cameras or other handheld devices may provide the way of engaging with adolescents and children. Electronic handheld devices may provide the opportunity to improve behaviors such as self-monitoring, in particular through the use of Short Message Service (SMS), or text messaging. For example, sending text messages to mobile telephones increased the effectiveness of a smoking cessation intervention among college students [12]. Another study among young adults in New Zealand revealed that participants who received text messages were more likely to quit smoking at 6 weeks compared to a control group [13]. In a program conducted among youth with type 1 diabetes [14], daily text messages were helpful for disease self-management, increased self-efficacy, and treatment adherence. In a randomized controlled trial of an Internet and mobile telephone-based physical activity intervention among overweight adults that included reminders for exercise sessions sent via cell phone, experimental partic-

ipants engaged in over 2 hours more physical activity per week than those with no access [15].

Mobile devices (e.g. a mobile telephone or PDA-like device) have evolved to meet market demand for general purpose mobile computing. Recently many mobile device based services and applications have emerged to help monitor nutrition habits of users and patients as possible substitutes for the conventional paper-and-pencil methods. Some of these applications have server side support in order to extend the range of features offered and also to partially retrieve the user's burden. In general, we can divide these services into three main groups. The first group is comprised with applications that offer basic food diary record capabilities where the user keeps an electronic record of foods and beverages consumed [16–21]. There is a second group where applications have a link that connects them with physical activity measures such as the number of calories expended depending on the physical activity and intensity [22–27]. Finally, there is a third group that offers the possibility of keeping visual records of food intake by using the integrated camera available in mobile devices [28–30]. Only [29] analyzes the images to estimate the energy content. It is not know how the images are analyzed (i.e. whether computer methods are used or only a human analyst). In general, the above applications use the mobile devices as instruments for field data collection rather than analysis.

As will be described later in this chapter a team at Purdue University and the University of Hawaii has combined the use of a camera on a mobile device with automated analysis of the images to design and deploy a complete dietary assessment system. One of the requirements of this system is the ability to automatically determine the type and quantity of food present in an eating occasion image.

1.1.3 Image Analysis For Food Recognition

In past few years methods for automatic food identification have gained some attention in the field of agriculture. Tillet in [31] gives a review of potential opportu-

nities for image analysis for agricultural processes. In 1977, Parrish and Goksel [32], developed the first food recognition system to detect apples. The system they proposed consisted of a camera, an optical filter, and the use intensity data to perform the analysis. After thresholding the image, the difference between the lengths of the horizontal and vertical extrema was used to determine "roundness". Radius and centroid information was also estimated from each thresholded region in the image. Finally, the density of the region was estimated by placing a mask, whose size was determined by the mean value of the extrema, on the centroid. In case the estimated density of the region was larger than a certain threshold, the region was said to be an apple.

In [33], a system to identify and locate green tomatoes was described. As the system proposed by Parrish and Goskel, the identification of tomatoes was based on shape information. The Circular Hough Transform (CHT) was used to determine binary edges and direction images.

In [34], a stick growing and merging method to segment complex food images was discussed. The sticks ends correspond to edge points. These sticks are horizontal lines in the image resulting from edge-preserving smoothing operations on the original image. Once the sticks are built, then adjacent sticks are merged to form sub regions. A boundary modification step was used to reduce the degree of boundary roughness of such sub regions. This approach was used to segment foods such as pizza, apples, pork and potatoes.

In 1988, a method to recognize food items using intensity-based methods was described using color filters to increase the contrast between the foods and the background [35]. The method consists of five steps: 1) image thresholding, 2) smoothing by a binary filter, 3) 8-neighbor connected component labeling for image segmentation, 4) feature extraction including area, perimeter, and compactness elongation information, and 5) classification by a nearest neighbor method. A similar approach was taken in [36], where the intensity-based methods and morphological operations

are combined. Harrel et al., in [37] presented a method for estimating the size and position of fruit which contained an initial valid pixel.

With the specific goal of dietary assessment, others have proposed various approaches for automatic food identification. In [38], a bootstrap procedure for selecting features from different feature channels is described. Voice annotation is used to constrain the number of candidate food classes (constrained set) from which the classifier has to choose. A pairwise classification framework is proposed where each category in the constrained set is visually pairwise compared with the others.

In [39], basic features to visually characterize foods included intensity, color, shape, and texture. The classification is based on a minimum distance classifier using statistical and co-occurrence features derived from the wavelet sub-bands. The method was tested on 15 food classes. The importance of the feature extraction step in automatic food recognition was discussed in [40–42] where several color, texture, and shape features were proposed.

In [43] a method for food identification that exploits the spatial relationship among different ingredients (such as meat and bread in a sandwich) is described. The food items are represented by pairwise statistics between local features of the different ingredients of the food items.

In [44], a multiple kernel learning method is described to integrate three sets of features namely color, texture, and Scale Invariant Feature Transform (SIFT) descriptors. All three features are fused together forming one single feature vector by assigning different weights to combine them for the final class decision. In [45], a system using video data recorded in restaurants is presented. SIFT descriptors are used to identify the food. In [46], an online food-logging system is presented, which distinguishes food images from other images, analyzes the food balance, and visualizes the log. Finally, in [47], a multi-view food classifier was described as part of a food intake assessment system. The approach consists of separating every food item through evaluating the best perspective (camera viewpoint) and recognize food items from multiple images.

1.2 Object Recognition

Object recognition can be divided into two broad groups: instance recognition and class recognition [48]. Instance recognition consists of identifying a particular known object being observed in a scene, surrounded by other objects, partially occluded and under various viewpoint conditions (e.g., automatic recognition of the painting *Mona Lisa* in different images). Class recognition aims at building a set of learning models in order to assign an unknown object to a class or category (e.g., recognize any painting and assign it to class *artwork*). Although both problems are sensitive to viewpoint changes, lighting conditions and other noise effects, class recognition is also sensitive to the variability existing within an object class [49]. In the rest of this thesis, we refer to class recognition as *object classification*.

Object classification is sometimes combined with segmentation. The goal of segmentation is to locate objects in an image. State-of-the-art segmentation methods use color and texture information to obtain image partitions by grouping together regions (segmented regions) of the image with homogenous color and/or texture appearance. Segmented regions can be somewhat unstable and only partially include the object. The goal of object recognition is to visually characterize the segmented region correctly and classify it as a part of an object according to a learning model so that multiple segmented regions of an object can be grouped together.

Schemes for visual object classification usually proceed in two stages. First, features of the object are measured or "extracted" and then the features are classified to obtain a decision regarding which class label to assign to a particular object [50].

1.2.1 Feature Description for Object Characterization

Object characterization refers to the representation of an object by features (visual properties) such as color, texture, shape, and/or edge information. Our goal is to determine features that can efficiently distinguish between objects belonging to different classes (inter-class discrimination), and also describe as much information

as possible of one class so that two objects of the same class, with different visual properties, can be classified together (e.g. a red chair and a blue chair should both be classified as *chairs*), this is known as intra-class robustness. The features need to also be robust to image perturbations. In general, we can group image perturbations into three categories: illumination changes, viewpoint changes, and noise effects [51].

- Illumination changes: a model to characterize changes in illumination by introducing a diagonal mapping with and offset factor was proposed Finlayson et al. in [52]:

$$\mathbf{I^c} = D^{u,c}\mathbf{I^u} + \mathbf{o} \tag{1.1}$$

  where $\mathbf{I^u}$ is the image acquired under an unknown light source u, $\mathbf{I^c}$ is the image transformed so it appears as if it was acquired using the reference light, c. $D^{u,c}$ represents a diagonal matrix mapping colors obtained by the unknown light source, u, to their corresponding colors under the reference light illuminant with a "diffuse" light $\mathbf{o}$:

$$\begin{bmatrix} I_R^c \\ I_G^c \\ I_B^c \end{bmatrix} = \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & c \end{bmatrix} \times \begin{bmatrix} I_R^u \\ I_G^u \\ I_B^u \end{bmatrix} + \begin{bmatrix} o_1 \\ o_2 \\ o_3 \end{bmatrix} \tag{1.2}$$

  Based on this model there are five types of common changes in the image values $I(x)$ [53]. First, *light intensity change* when $(a = b = c)$ and $(o_1 = o_2 = o_3 = 0)$. Second *light intensity shift*, where $(a = b = c = 1)$ and $(o_1 = o_2 = o_3)$. Third, *light intensity change and shift* when $(a = b = c)$ and $(o_1 = o_2 = o_3)$. Fourth, *light color change* when $(a \neq b \neq c)$ and $(o_1 = o_2 = o_3)$. Finally, *light color change and shift* when $(a \neq b \neq c)$ and $(o_1 \neq o_2 \neq o_3)$.

- Viewpoint changes: typically this includes variants of perspective transformations namely rotation changes, scale differences, and affine transformations. A viewpoint change is modeled by a transformation T that can be defined as:

$$T = \{\mathbf{I} : \exists\theta \,\text{such that}\, \mathbf{I} = \mathbf{T}_\theta\} \tag{1.3}$$

where $\mathbf{I}$ is the original image, and $\mathbf{I} = \mathbf{T}_\theta$ is the mapping function that models the viewpoint change.

In any given viewpoint transform the overlay luminance at pixel (x_1, y_1) of image I_1 should retain the luminance $L(x_1, y_1)$ of the pixel (x_2, y_2) in the transformed image I_2. (x_2, y_2) and (x_1, y_1) can be related by one of the following transformations:

- **The Translation Model**

$$\begin{aligned} x_2 &= x_1 + t_x \\ y_2 &= y_1 + t_y \end{aligned} \tag{1.4}$$

  where t_x, t_y are the displacement parameters;

- **The Rigid Model**

$$\begin{aligned} x_2 &= x_R \cdot \cos\alpha - y_1 \cdot \sin\alpha + t_x \\ y_2 &= x_R \cdot \sin\alpha + y_1 \cdot \cos\alpha + t_y \end{aligned} \tag{1.5}$$

  where α is the rotational angle (in radians);

- **The Rigid and Scale Model**

$$\begin{aligned} x_2 &= r \cdot [x_1 \cdot \cos\alpha - y_1 \cdot \sin\alpha] + t_x \\ y_2 &= r \cdot [x_1 \cdot \sin\alpha + y_1 \cdot \cos\alpha] + t_y \end{aligned} \tag{1.6}$$

  where r is the scaling factor;

- **The Affine Model**

  * 6 parameter affine transformation

$$\begin{aligned} x_2 &= a_0 \cdot x_1 + a_1 \cdot y_1 + t_x \\ y_2 &= a_2 \cdot x_1 + a_3 \cdot y_1 + t_y \end{aligned} \tag{1.7}$$

  * 8 parameter affine transformation or perspective model

$$\begin{aligned} x_2 &= \tfrac{(a_0 \cdot x_1 + a_1 \cdot y_1 + a_2)}{(a_6 \cdot x_1 + a_7 \cdot y_1 + 1)} \\ y_2 &= \tfrac{(a_3 \cdot x_1 + a_4 \cdot y_1 + a_5)}{(a_6 \cdot x_1 + a_7 \cdot y_1 + 1)} \end{aligned} \tag{1.8}$$

- Noise effects: noise can model many image perturbation effects. Two major sources of noise effects are particularly critical in our work, namely shadows and reflections. Shadows are formed whenever an occlusion partially blocks the illumination of the surface being imaged; this is an inseparable aspect of all natural scenes. Image analysis is a difficult task and shadows often interfere with object classification. Both shadows and reflections may change the object visual appearance and therefore sometimes pre-processing techniques are required to remove reflections and shadows from the image.

1.2.2 Object Classification

Based on the object features, the classification system selects the most likely category or class label. Many classification approaches are based on the "parts and structure" model introduced by Fischler and Elshlager [54], where the object is classified by "finding its constituent parts and measuring their geometric relationships". This model has been extended to detect and learn new classes [55]. Another approach for object labeling is to represent each object as "unordered collections of feature descriptors" creating what is known as a *Bag of Features (BoF)* or a *Bag of Words* [51]. This can be seen as an attempt to represent the object by the characterization of its atomic components (local features).

Methods have been developed for classification by linear separation (e.g. class and non-class object separation) in the feature space using an hyperplane. Unfortunately, in many cases, the representation of the features projected into the feature space does not allow such simple separation, and finding the hyperplane is difficult to accomplish. One approach to obtain better classification for non-linear features is to map the original feature space to a much higher dimensional space in which the classes become separable. This can be accomplished by using a Support Vector Machine (SVM) with nonlinear kernels [56]. However, these approaches are very dependent on the nature of the features. For example, food samples can dramatically vary in appearance. Such

variations may arise not only from changes in viewpoint and illumination, but also from non-rigid deformations, and variability in shape, texture, color and other visual properties.

A different approach to the above is to only project a subset of the original features. Strategies of (optimal) selection of features suggest a trade-off between the complexity of the classification scheme and the complexity of features [57]. However, methods using simple generic features in very high dimensional spaces usually are combined with elaborate classification schemes [58]. Sometimes the selection of one approach over another depends on the type of application, and thus on the dataset. For a small number of classes creating specific features and selecting a simple space separation scheme may be more appropriate. On the contrary, for large number of classes, elaborate classifiers combined with grammar rules may be more effective.

Our objective is to find informative features where simple classification approaches can be used. Multiple feature spaces can provide distinct sets of information to increase the classification efficiency. In a multichannel feature classification framework, one final decision needs to be obtained by combining multiple feature spaces. To combine feature spaces, the Multiple Kernel Learning (MKL) paradigm is often used. MKL uses combined kernels which are a weighted combination of several kernels from different feature spaces [59]. This approach usually involves processing very high dimensional feature spaces, which increases the computational complexity of the system. Instead of fusing feature spaces, a combination of individual classifications can be performed, where a weighted scheme is used with each individual class decision.

Training data plays an important role in the success of object classification tasks. Generating models with class samples that do not possess features that visually stand out and represent such class may lead to bad classification performance. Also, a large number of classes can make the classification task more complex, and even intractable. One solution to this is to set constraints in the training data. In some applications, the number of classes can be reduced. For example, in our food classification problem,

having individual classifiers (one classifier per user) may result into small number of classes that the classifier has to select among, *i.e.* less class uncertainty.

After the classification of an object based on its appearance (features), contextual information can be used to refine the decision [60,61]. Context refers to any information that is not directly produced by the appearance of an object [61] (pixel values). Contextual refinement is often considered as 'side' information, and it is combined with part-based models (segmentation + classification) into the same system [62]. In the same direction recently, a number of approaches use the gist of a scene as contextual information [63]. The idea is to, first, identify the "topic" of the scene, (e.g. indoor vs. outdoor scene, birthday party vs. a funeral, etc.) and then refine the part-based classification model by selecting potential classes belonging to such topic and constraining the problem to less number of classes.

1.2.3 Feature Extraction and Classification Model

In the feature extraction and classification system described in this thesis, each training image, I_t, is partitioned into segmented regions, S_t, by using ground truth (manual segmentations) or automatic segmentation. Each training segmented region, S_t, corresponds to exactly one object (food item) of class $\lambda_i \in \Lambda$, where Λ is the set of all object class labels (food types, e.g. lettuce, soup, milk). These segmented regions are collected for all training images into the training set $\{S\}$. For each training segmented region $S_t \in S$, we use several types of features (e.g. color, texture, and local features), and within each type of feature we estimated different multidimensional statistics forming feature vectors f that describe data points in a feature space. The feature spaces projected by these feature vectors are referred to as *feature channels*. For example, for local features we investigated several feature channels, including *Scale Invariant Feature Transform (SIFT) features*, or *Speeded-Up Robust Features (SURF)* among others (Chapter 2). Once training feature vectors are constructed for each segmented region, we generate a learning model to classify testing segmented

regions, S_q, obtained from input images, I_q, (Chapter 3). The classification consists of three stages:

1. Individual feature channel classification - each feature channel is classified separately

2. Late fusion decision - we combine the decision from the classifiers for each feature channel

3. Contextual refinement - we correct the decisions based on contextual information (e.g. exploiting the fact that mashed potatoes with green beans occur frequently) [1]

1.2.4 Multiple Hypothesis Segmentation and Classification

Our feature extraction and classification model has been integrated into a multiple hypothesis segmentation and classification system (MHSC) [64]. The goal is to obtain "optimal" segmentations of the input image using the feedback from the classifier (object label and confidence score).

The segmentation step partitions the image producing multiple segmented regions. Many segmentation methods such as Normalized Cuts [65], use the number of segmented regions as one of the input parameters of the segmentation method. Since, the exact number of segmented regions in an image is not known *a priori*, a particular choice of the number of segmented regions results in either an under segmented or over segmented image. That is, some of the segmented regions may contain pixels from more than one class (over segmented) while more than one segmented region may correspond to a single class (under segmented). In order to obtain accurate segmentations of the image a joint iterative segmentation and classification system is described in [66, 67], where the classifier's feedback (i.e. class label and confidence score) is used to obtain a final "optimal" segmentation mask. Figure 1.1 describes

[1] Context refers to any information that is not directly produced by the visual appearance of an object. In this work we use object combination likelihood and misclassification rates as context.

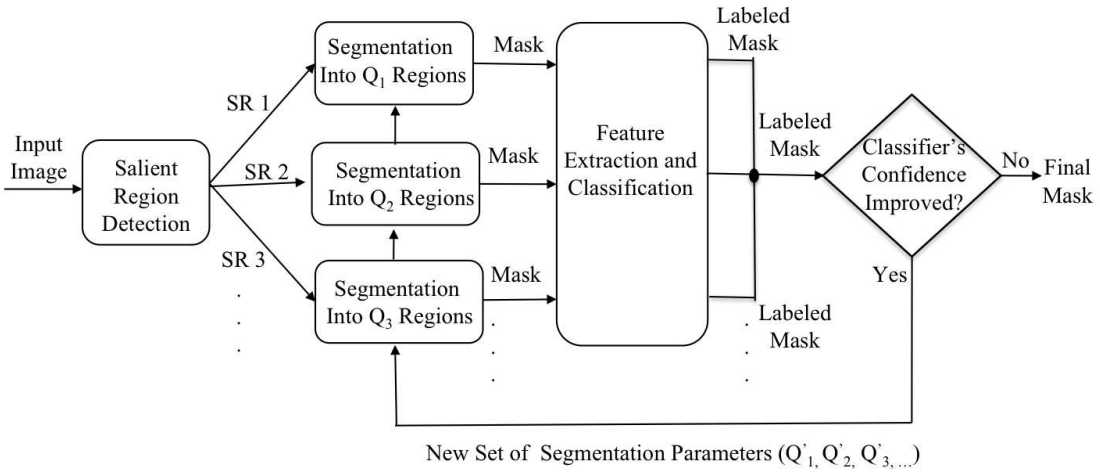


Fig. 1.1. Multiple Hypothesis Segmentation and Classification System.

our MHSC approach. Given an input image, we first use salient region detection to identify potential regions in the image containing objects of interest. For each salient region SR1, SR2, SR3, . . . , multiscale segmentation is performed. As a result of this operation, each salient region is partitioned into multiple segmented regions (e.g SR1 is partitioned into Q_1 regions, SR2 into Q_2 regions,...). For each resulting segmented region, feature extraction and classification is performed. The classifier assigns to each segmented region a class label and a confidence score that indicates the "classifier's confidence" that its inferred label is correct. This information is evaluated based on a stability criteria. If the stability condition is not satisfied, the system partitions each salient region, again, into a new number of segmented regions (e.g SR1 is partitioned into $Q_1^{'}$ regions, SR2 into $Q_2^{'}$ regions,...) and the classification process is repeated. Satisfaction of the stability condition indicates that the classifier is no longer changing its class label decision, and thus, the best possible segmentation of the image has been reached.

1.3 The TADA System

The importance of using an image food record approach versus classical food record approaches (handwritten records) is discussed in [11]. In this study 30 adolescents were asked to use a disposable camera and small notebook during a one week period for food intake recording. It was noted that 17 of 30 participants recorded in the notebook, whereas 29 of 30 took pictures. In the same study, adolescents shown a strong preference for using methods that incorporate technology such as capturing images of food.

In the Technology Assisted Dietary Assessment (TADA) project at Purdue University, we are developing imaging based tools in order to automatically obtain accurate estimates of what foods a user consumes. A step towards dietary assessment using electronic handheld devices is to make use of the integrated digital camera in mobile telephone to take images of food. The goal is to develop tools that reduce user burden while providing accurate estimates of energy and nutrient intake. We have developed a novel food record method using a mobile device and the embedded camera. This is known as Mobile Device Food Record (mdFR). Images acquired before and after foods are eaten can be used to estimate the amount of food consumed. This work is sponsored by grants from the National Institutes of Health as part of the Genes, Environment and Health Initiative. This project is the result of a large collaboration between various departments at Purdue University, the University of Hawaii, and the Curtin University of Technology in Australia.

The main stages of the image analysis in the TADA system are as follows:

- Image Calibration and Acquisition: this includes pre-processing the image in order to guarantee a minimal image quality by detecting blur levels and color correcting the image [68, 69].

- Image Segmentation: the goal is to locate and isolate the food in the eating occasion scene [66, 67, 70].

- Feature Extraction: this step consists of extracting visual characteristics of each segmented region obtained by the segmentation process [71, 72].
- Classification: the goal of the classification is to determine a category/label for each segmented region based on its features [66, 72].
- Volume Estimation: based on the segmentation, the category label, and reference size estimation the volume of food consumed is determined. Each food item has a 3D shape template associated to it that it is composite over the 2D segmentation mask of the food item [73–75].

The contributions of this thesis with respect to the TADA system are in the areas of feature modeling and classification and the development of the food image database systems.

1.3.1 TADA System Architecture

The TADA system consists of two main parts: a mobile application, we refer to as the Mobile Device Food Record (mdFR) and the "backend" system consisting of the computation server and database system. In the TADA system images captured "before" and "after" eating occasions are used to estimate the food intake. Each food item in the image is segmented, identified, and its volume is estimated [66, 72, 74]. From this information, the energy and nutrients consumed can be determined. Figure 1.2 shows an overview of our system. The first step is to use the mobile device (e.g. a mobile telephone) to acquire images of an eating occasion and send them to the server along with appropriate metadata (e.g. date, time, and geolocation). Automatic image analysis (image segmentation and food identification) is done on the server (step 2). These results are sent back to the users (step 3) where they review and confirm the analysis information (step 4). In step 5, the server receives the information back from the user. These results are used for final image analysis refinement and volume estimation. Step 6 consists of structuring the data generated

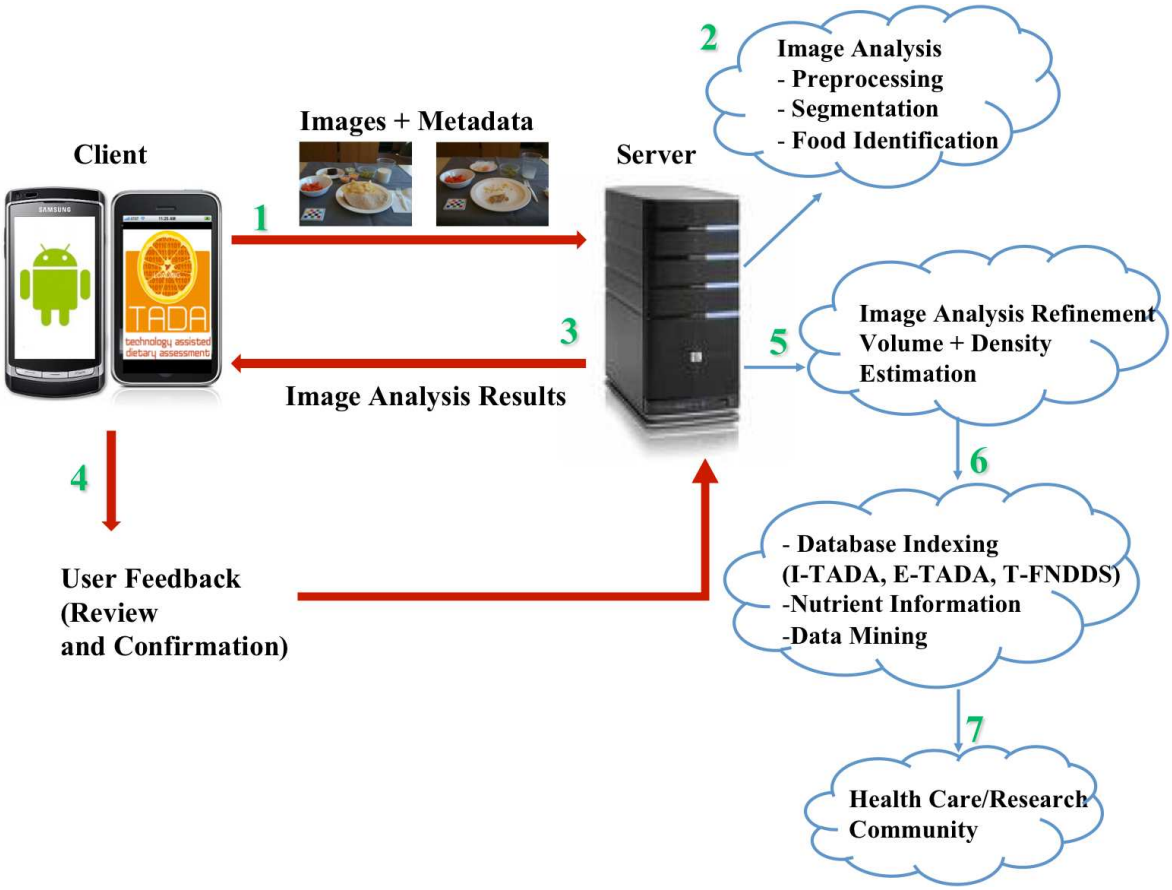


Fig. 1.2. The TADA System Architecture.

in the previous steps by forming object descriptions (e.g. user information, images, metadata, and image analysis results) and storing them in a database system. Also in step 6, nutrient information (e.g. calories consumed) is determined from an extended version of the USDA Food and Nutrient Database for Dietary Studies (FNDDS) database [76] [1]. Finally, these results are available for healthcare professionals in a web-based interface of the database system for further analysis (step 7).

One of the keys for success of this system is that there is a need to minimize the user efforts to monitor his/her nutritional status. The objective is to design an easy-to-use application where the user only needs to send the images to a server to

[1]The FNDDS is a database containing foods eaten in the U.S., their nutrient values, and weights for standardized food portions.

be analyzed, without having to connect his/her mobile device to any other computer-based device for analysis.

1.3.2 Integrated Food Image Databases

Recently, several "nutrition" applications and internet-based tools and services are available for the iPhone and Android devices. Some of them also include the availability to acquire eating occasion images. These applications include *Hyperfit* [27], *Calorie Counter* [26], *Meal Snap* [29] and *PhotoCalorie* [30]. There are also various food image databases available such as StockFood [77], Food Testing Image Database from Appealing Products, Inc. [78], and Food-Image.com [79]. While these applications are interesting, they usually provide little true nutrition information and it is not clear how they can be used by health care professionals for dietary assessment. Furthermore, it is not obvious how the food images some of them use are analyzed (i.e. whether automatic methods are used or only a human analyst is employed). In general these applications use the mobile devices as instruments for "data collection" rather than doing complete dietary analysis.

The TADA system generates large amounts of data: data generated by users, dietitians, and the image analysis that needs to be stored in a structured manner. Also, there is a need for a nutritional database in order to map the results of the image analysis to nutritional information. A food image database system has been developed to fulfill such needs.

An integrated database system has been developed in this thesis where images, users, and nutritional information are equally important. Three unique interconnected databases have been created in order to provide a platform for researchers to explore and discover embedded patterns in dietary habits, extend traditional nutritional databases including image and patient related data, and coordinate the data flow between all processes.

1.4 Contributions Of This Thesis

In this thesis an imaging based tool to assess diet is described. In particular, methods for automatic food identification and the design and deployment of an integrated food image-based database system have been investigated. We have focused our efforts on designing features for object detection in order to visually characterize objects in a scene. The main contributions in the area of food identification are as follows:

- We investigated several color features from various color space components. Local and global color features have been proposed in order to characterize many types of objects such as objects with homogeneous color distribution and complex objects composed by primary components (ingredients) with different color distributions. The color features have shown to be robust to color perturbations that increase intra-class variance (e.g. green vs. ripe fruit).

- We proposed three unique texture descriptors namely the Entropy-based categorization and Fractal Dimension estimate (EFD), the Gabor-based image decomposition and Fractal Dimension estimate (GFD), and the Gradient Orientation Spatial-Dependence Matrix based on the spatial relationship of gradient orientations (GOSDM).

- We examined local descriptors in order to form visual vocabularies.

In the classification process we have proposed a multichannel classification system. The main contributions are as follows:

- We investigated late decision fusion in order to combine multiple feature channels as opposed to fusing feature spaces into a higher dimensional feature space.

- We derived confidence scores for SVM and KNN classifiers in order to combine multiple feature spaces into one class decision. These proposed scores have also been used to improve the image segmentation.

- We explored the use of contextual information to refine the classifier decision by developing a model to obtain a labeling agreement based on object interaction and misclassification rates.

We also proposed an integrated food image-based database system for our mobile device food record system. We extended our database by developing a web-based application that allows researchers to monitor patients in real-time. We enhanced traditional nutritional databases with data obtained from food images as follows:

- We proposed an integrated food image-based database system where data from users and images is connected to nutritional information for dietary assessment.
- We created a web-based interface for nutritionists, dietitians, and researchers for guidance and monitoring of patients.
- We extended the USDA FNDDS to include image related data.
- We created a cooperative platform to explore and discover embedded patterns in dietary habits.

1.5 Publications Resulting From This Work

Journal Articles:

1. **Marc Bosch**, Nitin Khanna, Carol J. Boushey, and Edward J. Delp, "An Integrated Image-Based Food Database System with Application in Dietary Assessment," *IEEE Transactions on Information Technology in Biomedicine*, submitted.
2. Fengqing Zhu, **Marc Bosch**, Nitin Khanna, Carol J. Boushey and Edward J. Delp, "Multiple Hypothesis Image Segmentation and Classification with Application to Dietary Assessment," *IEEE Transactions on Image Processing*, submitted.

3. Fengqing Zhu, **Marc Bosch**, Insoo Woo, SungYe Kim, Carol J. Boushey, David S. Ebert, Edward J. Delp, "The Use of Mobile Devices in Aiding Dietary Assessment and Evaluation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 4, pp. 756-766, August 2010.

4. Bethany L. Six, TusaRebecca E. Schap, Fengqing Zhu, Anand Mariappan, **Marc Bosch**, Edward J. Delp, David S. Ebert, Deborah A. Kerr, Carol J. Boushey, "Evidence-Based Development of a Mobile Telephone Food Record," Journal of American Dietetic Association, January 2010, pp. 74-79.

Conference Papers

1. Fengqing Zhu, **Marc Bosch**, Ziad Ahmad, Nitin Khanna, Carol J. Boushey, Edward J. Delp, "Challenges in Using a Mobile Device Food Record Among Adults in Freeliving Situations," *mHealth Summit*, December, 2011, Washington D.C.

2. **Marc Bosch**, Fengqing Zhu, Nitin Khanna, Carol J. Boushey, Edward J. Delp, "Combining Global and Local Features for Food Identification and Dietary Assessment," *Proceedings of the IEEE International Conference on Image Processing*, Brussels, Belgium, September 2011.

3. Fengqing Zhu, **Marc Bosch**, N. Khanna, Carol J. Boushey, Edward J. Delp, "Multilevel Segmentation for Food Classification in Dietary Assessment," *Proceedings of the International Symposium on Image and Signal Processing and Analysis*, Dubrovnik, Croatia, September 2011.

4. **Marc Bosch**, Fengqing Zhu, Nitin Khanna, Carol J. Boushey, Edward J. Delp, "Food Texture Descriptors Based on Fractal and Local Gradient Information," *Proceedings of the European Signal Processing Conference (Eusipco)*, Barcelona, Spain, August 2011.

5. **Marc Bosch**, TusaRebecca E. Schap, Nitin Khanna, Fengqing Zhu, Carol J. Boushey, Edward J. Delp, "Integrated Databases for Mobile Dietary Assess-

ment and Analysis," *Proceedings of the 1st IEEE International Workshop on Multimedia Services and Technologies for E-Health in conjunction with the International Conference on Multimedia and Expo (ICME)*, Barcelona, Spain, July 2011.

6. Fengqing Zhu, **Marc Bosch**, Nitin Khanna, TusaRebecca E. Schap, Carol J. Boushey, David S. Ebert, Edward J. Delp, "Segmentation Assisted Food Classification for Dietary Assessment", *Proceedings of Computational Imaging IX, IS&T/SPIE Electronic Imaging*, San Francisco, CA, January 2011.

7. SungYe Kim, TusaRebecca E. Schap, Marc Bosch, Ross Maciejewski, Edward J. Delp, David S. Ebert, and Carol J. Boushey, "A Mobile User Interface for Image-based Dietary Assessment", *Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia*, Limassol, Cyprus, December 2010.

8. **Marc Bosch**, Fengqing Zhu, TusaRebecca E. Schap, Carol J. Boushey, Deborah Kerr, Nitin Khanna, Edward J. Delp, "An Integrated Image- Based Food Database System with Application in Dietary Assessment," *mHealth Summit*, November, 2010, Washington D.C. *(Meritorious New Investigator Award)*

9. Fengqing Zhu, **Marc Bosch**, Carol J. Boushey, and Edward J. Delp, "An Image Analysis System for Dietary Assessment and Evaluation," *Proceedings of the IEEE International Conference on Image Processing*, September, 20010, Hong Kong.

10. Anand Mariappan, **Marc Bosch**, Fengqing Zhu, Carol J. Boushey, Deborah A. Kerr, David S. Ebert, Edward J. Delp, "Personal Dietary Assessment Using Mobile Devices," *Proceedings of the IS&T/SPIE Conference on Computational Imaging VII*, Vol. 7246, San Jose, January 2009.

2. FEATURES FOR OBJECT CHARACTERIZATION

2.1 Features for Visual Characterization

Recognition is one of the most important functions of the human visual system. As humans, we can recognize objects, materials, surface properties, and scenes by simply observing without touching. We are able to recognize both individual objects (Instance recognition), as well as categories of objects (class recognition). According to [80], by the time we are six years old we are capable of recognizing more than 104 categories of objects. As we learn, our brain organize both objects and object categories into useful and informative entities and relate them to language.

An essential step in solving any object classification problem starts by determining visual characteristics that are salient of the object's visual appearance and can be used to separate that object from other objects in the image. This is commonly known as *feature modeling*. It consists of estimating a set of statistical parameters (low-level features) from the object or parts of the object. These features are grouped into feature vectors, and most of the time there is only one feature vector available per object or segmented region, which is used to determine the class label. Features that represent the object with one feature vector are often referred to *global features*.

Another type of low level features have been investigated namely *local descriptors* based on multi-scale or scale-space representation. The main difference with global features is the size of the region used to estimate such features. In the early eighties Witkin [81] proposed to consider scaling to formulate the principal rules of modern scale-space theory relating image structures represented at different scales. In his approach the scaling parameter is taken as a continuous parameter over several resolution (scale) representations of the same image. Other contributions have been made by Koenderink [82], Lindeberg [83] and Florack [84] in scale-space theory. Low-level

features exist only within a limited range of scales between the inner and outer scale, that is the smallest and the largest scale of interest respectively (e.g. the concept of a tree only makes sense at scale from few centimeters to few meters). Scale-space theory provides tools to describe the image following a multi-scale signal representation and then find characteristic features at particular scales depending on the class. Works proposed in [51, 85, 86] highlighted the importance of using local features for object classification.

Our goal is to find features that describe an object so that we can distinguish between different types of objects while classifying different samples of the same object class that has the same category/label. In this thesis we investigated many features to assess the role of each type of feature for visual description of a food item. In this chapter, we describe the features that we have investigated. Our current approach uses both global and local features. By global features we mean features that describe the entire segmented region with a single feature vector. Local features are designed to describe local visual characteristics around salient or invariant points in the segmented region. In early experiments we found that global features are not sufficient and can be of limited use to discriminate between different foods when there is object occlusion, change of pose, lighting variations, or for segmented regions that represent objects composed by multiple components (ingredients). In our system, features are extracted/measured after the segmentation step [66].

2.2 Global Features

Typically, color, texture, edge, and shape information are the four types of object representations most widely used in order to describe global characteristics of an object. However, segmentation, in general, does not preserve the global shape information of an object due to unstable segmentations. By unstable segmentations we mean that an object is segmented based on both the object structure and on the overall image or scene structure [66, 67]. This is frequent in segmentation approaches

where the number of segmented regions is fixed. In addition to the problems associated with segmentation, many foods have large variations in shape due to eating conditions. Therefore, shape features are not used in our system. Edge information usually is represented by edge histograms such as edge direction histograms that capture the spatial distribution of edges, and edge intensity histograms which measure the degree of uniformity of the edge pixels [87]. We believe that appropriate texture features can capture more information than provided by edge descriptors with similarly computational complexity. Therefore, in this thesis only color and texture descriptions were used as global features.

2.2.1 Color Features

Color is an important discriminative property of food, allowing us to distinguish, for example, between mustard and mayonnaise, and in some cases it is the only information available to distinguish between liquids (e.g. orange juice and milk). There exist many color spaces, most consisting of three-dimensional color representations. In the human visual system there are three different types of color sensitive receptors corresponding to the primary red, green, and blue colors [88]. The *Commission International de l'Eclairage* (CIE) standardized the primary colors at wavelengths $\lambda_R = 700nm$, $\lambda_G = 346.1nm$, and $\lambda_B = 435.8nm$ in 1931 [89]. RGB color representation has become the standard for image storage formats.

Another widely used color space is *Lab*, which is a color-opponent space with dimension L^* for lightness and a^* and b^* for the color-opponent dimensions. $L^*a^*b^*$ color is designed to approximate human vision. This corresponds closely to perceptual uniformity and its L component closely matches human perception of lightness. Finally, *HSV* (Hue, Saturation, Value) color space corresponds to human perception, although it is not perceptually uniform, *i.e.* the separation in color spaces is not proportional to the human visual system color dissimilarity.

Color information is very sensitive due to color variations in natural scenes, such as changes due to shadows, and light source reflections. One approach to understanding color is through the use of the dichromatic reflection model by Shafer [90]. The model describes how photometric changes, such as shadows and other specularities, affect RGB pixel values. In order to reduce such variations in illumination, one can use color descriptors that are approximately invariant to illumination changes [91].

Food has large variations of color. Fresh food may contain different color chromaticities relative to their ripeness. Whether they are consumed raw or cooked light reflections can produce different effects and chromaticities. Although there are many manufactured foods that have very homogeneous color properties, there are also a large number of foods with many colors due to the ingredients in the food composition. Hence, there is no unique feature or color description that can be used to characterize food. In order to address many of the color effects found in food, we considered three types of color features.

These are the three global color feature channels used in our system:

- Global color statistics
- Entropy color statistics
- Predominant color statistics

In this thesis we use the term color channels to describe each of the three above global color feature spaces (Global color statistics, Entropy color statistics, Predominant color statistics). We use the term color space components to describe each of the individual elements that compose each color space (e.g. R, G and B are color space components of the RGB color space).

Global Color Statistics

Mindru *et. al.* proposed a set of color moments, $M_{p,q}^{a,b,c}$, invariant to changes in lighting conditions [92]. Let I define RGB triplets for image coordinates (x, y),

$I : (x, y) \rightarrow (R(x, y), G(x, y), B(x, y))$, and by using RGB triplets as a distribution, it is possible to define moments as follows [92]:

$$M_{p,q}^{a,b,c} = \int\int x^p y^q [I_R(x, y)]^\alpha [I_G(x, y)]^\alpha [I_B(x, y)]^\alpha \tag{2.1}$$

$M_{p,q}^{a,b,c}$ is known as the generalized color moment of order $p + q$ and $a + b + c$. Based on this, we estimated a set of moments from several color space components. We call this color descriptor the *global color statistics*. We only considered the 1st, and 2nd 1-band moments. These are moments involving only a single color space component. These moments are obtained for different color spaces, the question becomes what color spaces capture more discriminant information for object classification. We selected $R, G, B, Cb, Cr, a^*, b^*, H, S, V$ space components. *R, G, B* were selected because many imaging sensors represent color information in the RGB color space. Using the original color values would avoid the introduction of undesired transformation effects. *Cb, Cr* are approximations to color processing and perceptual uniformity. a^*, b^* are approximately perceptually uniform and try to mimic the logarithmic response of the human visual ssyem. *H, S, V* channels were chosen because they correspond closely to human perception. The *Global color statistics* feature is an attempt to characterize the object by capturing its average color composition.

Cb, Cr are two color components of the YCbCr color space. YCbCr color space can be obtained from RGB triplets by the following transformation:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.500 \\ 0.500 & -0.419 & -0.081 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.2}$$

where R, G, B are the pixel values in the RGB color space.

a^*, b^* color components can be obtained from the XYZ space components as follows:

$$\begin{aligned} L^* &= 116 f\left(\tfrac{Y}{Y_w}\right) - 16 \\ a^* &= 500\left(f\left(\tfrac{X}{X_w}\right) - f\left(\tfrac{Y}{Y_w}\right)\right) \\ b^* &= 200\left(f\left(\tfrac{Y}{Y_w}\right) - f\left(\tfrac{Z}{Z_w}\right)\right) \end{aligned} \tag{2.3}$$

where

$$f(x) = \begin{cases} x^{\frac{1}{3}} & x > 0.008856 \\ 7.787x + \frac{16}{116} & otherwise \end{cases} \tag{2.4}$$

where X_w, Y_w, Z_w are the CIE XYZ tristimulus values of the reference white point [93], the function $f(x)$ is divided into two domains to prevent the infinite slope at $x = 0$. X, Y, Z values can be obtained from RGB by the following operation [94]:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \frac{1}{0.17697} \begin{bmatrix} 0.49 & 0.31 & 0.20 \\ 0.17697 & 0.81240 & 0.01063 \\ 0.00 & 0.01 & 0.99 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{2.5}$$

Finally, *H,S,V* are found by first finding the maximum (max) and minimum (min) from the RGB triplet. Saturation S, is then:

$$S = \frac{(max - min)}{max} \tag{2.6}$$

and Value, V, becomes:

$$V = max \tag{2.7}$$

The Hue, H, is estimated by first calculating $R^{'}, G^{'}, B^{'}$:

$$\begin{aligned} R^{'} &= \frac{max - R}{max - min} \\ G^{'} &= \frac{max - G}{max - min} \\ B^{'} &= \frac{max - B}{max - min} \end{aligned} \tag{2.8}$$

and then examining the following:

$$\begin{aligned} if\ R = max\ and\ G = min \quad then \quad & H = 5 + B' \\ elseif\ R = max\ and\ G \neq min \quad then \quad & H = 1 - G' \\ elseif\ G = max\ and\ B = min \quad then \quad & H = R' + 1 \\ elseif\ G = max\ and\ B \neq min \quad then \quad & H = 3 - B' \\ elseif\ R = max \quad then \quad & H = 3 + G' \\ otherwise \quad & H = 5 - R' \end{aligned} \tag{2.9}$$

Finally, H, Hue is converted to degrees by multiplying by 60. S and V have values between 0 and 1, and H between 0 and 360 [95].

Entropy Color Statistics

This feature is used to characterize the distinctiveness and repeatability of color information for each color component. There are foods and objects composed of many colors so that by using an average color we cannot fully represent the color composition. Figure 2.1 shows an example of two food samples of the same class (*vegetable soup*). One way to address this is by the color distinctiveness of the object. We propose measuring color distinctiveness of a color space component by using entropy [96]. For each pixel in the segmented region, the entropy is estimated using a $N \times N$ pixel neighborhood, where N is a function of the horizontal (H) and vertical (V) dimensions of the segmented region, defined as $N = min(max(min(H, V)/4, 16), H, V)$, with a minimum value of N equal to 16 pixels. The final feature vector is formed by estimating the 1st and 2nd moment statistics of the entropy of the R, G, B color space components. Given a pixel x and a local neighborhood M_p, we can estimate the pixel entropy H_x for each color space component as [96]:

$$H_x = -\sum_n p_{x,M_p} \log_2 p_{x,M_p} \tag{2.10}$$

Fig. 2.1. Two Samples Of The Same Food (*Vegetable Soup*) With Different Color Distributions.

where n is the pixel range, *i.e.* $(n = 0, ..., 255)$, and p_{x,M_p} is the probability of a pixel (in M_p) for each color space component.

Predominant Color Statistics

It was shown in [97] that the human visual system cannot simultaneously perceive a large number of colors. The number of colors that can be represented in cognitive space is approximately 30 [98, 99]. Most natural scenes can be described by forming a compact color representation using only a few colors to describe their color content with a minimal color quality penalty [100]. Based on this, our *predominant color statistic* descriptor encodes the distribution of the salient colors in the object by selecting the P most representative colors (in RGB space) for a segmented region. In this context, salient colors refers to the most dominant colors in the object. In our implementation the RGB space is quantized into a 1000-cell cube where each cell represents a RGB color triplet. This cube can be seen as a 1000-bin histogram. The P largest peaks of the color histogram are considered to be the predominant colors.

Figure 2.2 shows examples of representing food items with the predominant colors ($P = 4$). The final feature for this color descriptor is defined as:

$$F = \{(c_1, p_1, v_1), ..., (c_P, p_P, v_P)\} \tag{2.11}$$

where c_i represents the 3-D color vector from the RGB cube, p_i is the percentage of color in the total object, and v_i is the color variance inside the region described by the predominant color. The total dimension of the feature vector is $(7 \times P)$. A similar color descriptor is used in the MPEG-7 standard, known as Dominant Color Descriptor (DCD) [87].

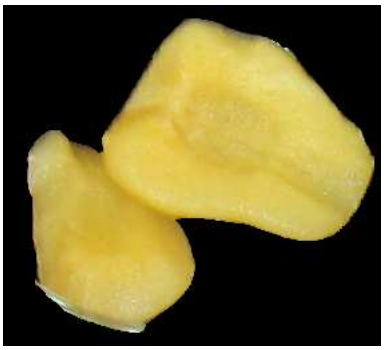
2.2.2 Texture Features

For many objects, texture is a very descriptive feature. In general, texture describe the "arrangement of basic elements of a material on a surface" [101, 102]. Typically four approaches have been used for texture features [103]: statistical methods, model-based methods, transform or spatial-frequency methods and structural methods [104,105]. Statistical methods represent textures by non-deterministic properties that model the relationships of the gray levels of the image [106]. Model-based texture analysis can use fractal dimension to describe the irregularities of the texture surface [107], or can use stochastic models [108, 109]. Transform methods, such as Fourier, Gabor and wavelet transforms, measure local texture characteristics in the frequency domain [110–112]. Structural methods represent textures by primitives (microtextures) and spatial arrangements (macrotextures) of the primitives [113, 114]. Local features and kernels have been proposed for texture and object classification [115,116]. Many of these local descriptors are based on gradient orientation. How one effectively models the spatial relationship among local features across different samples of the texture class is still an open problem.

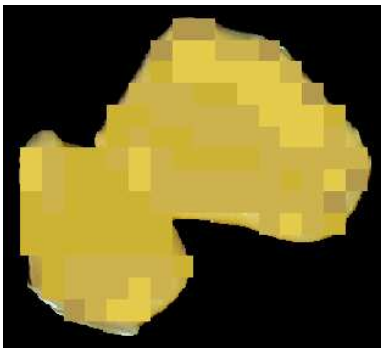
In this section we describe three texture descriptors for food classification. Our first texture descriptor, *Gradient Orientation Spatial-Dependence Matrix (GOSDM)*, is based on the occurrence rate of the spatial relationship of gradient orientations

(a) Spaghetti

(b) Predominant Color of Spaghetti

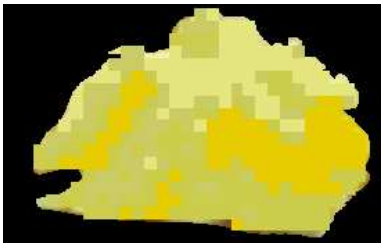


(c) Pears



(d) Predominant Color of Pears

(e) Scrambled Eggs



(f) Predominant Color of Scrambled Eggs

Fig. 2.2. Examples Of Food Items Represented Using Predominant Colors. (a,b) Spaghetti. (c,d) Pears. (e,f) Scrambled Eggs.

for different neighborhood sizes. Using fractal information theory we propose an *Entropy-based categorization and Fractal Dimension estimation (EFD)*, and a *Gabor-based image decomposition and Fractal Dimension estimation technique (GFD)*. In section 4.2 we present a series of results to evaluate the performance of our texture features and compare them with well-known texture features namely *Gray Level Co-*

occurrence Matrix (GLCM) [117], Gabor features (Gabor) [111, 118] and *Multifractal Spectrum (MFS)* [119].

Gradient Orientation Spatial-Dependence Matrix (GOSDM)

Haralick in [117] suggested the use of GLCMs for texture analysis purposes, assuming that the texture information is contained in such matrix. In the late 80s, extensions of GLCMs were proposed by [120, 121] focusing on faster implementations of the GLCM.

Based on the GLCM, we propose the GOSDM where instead of using gray level pixel interactions we consider gradient orientation interactions. The probability of occurrence between pairs of gradient orientations is estimated by constructing a series of GOSDMs. The use of gradient information as opposed to gray level values is motivated by the fact that gradient information is more robust to illumination changes and other distortions with respect to gray level pixel values. If we consider two identical texture patterns with different gray tones (illumination effects), the GLCM will increase the intra-class variation while the GOSDM retains more structural information while removing some redundancy in the gray-level spatial relationship. This effect is illustrated in Figure 2.3 where there are two samples of the same texture pattern with different gray scale levels. For a specific offset value the GLCM estimated from Figure 2.3.a will be different from the GLCM determined from Figure 2.3.b. On the contrary, the GOSDMs estimated from the texture shown in Figure 2.3.a and Figure 2.3.b will remain the same. Thus, GLCMs are more sensitive to the gray level variations within the same patterns, creating noisier feature spaces.

There are 5 main steps involving the estimation of GOSDM features:

- Gradient orientations estimation
- Dominant direction estimation
- Gradient orientation spatial dependence matrix estimation

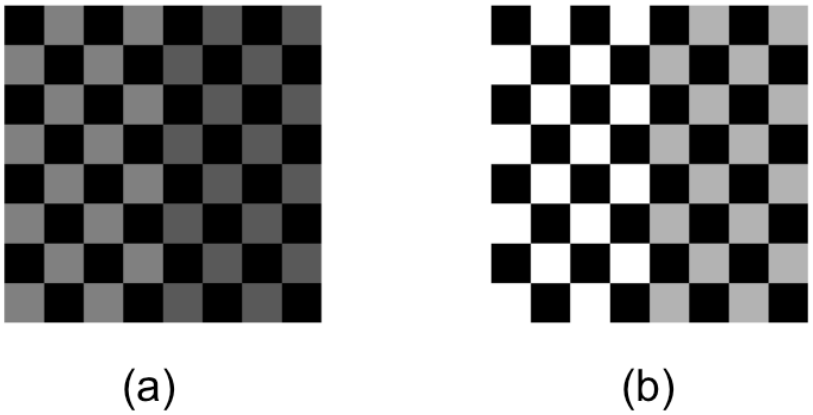


Fig. 2.3. Two Samples Of The Same Texture Only Differing By Their Gray Level.

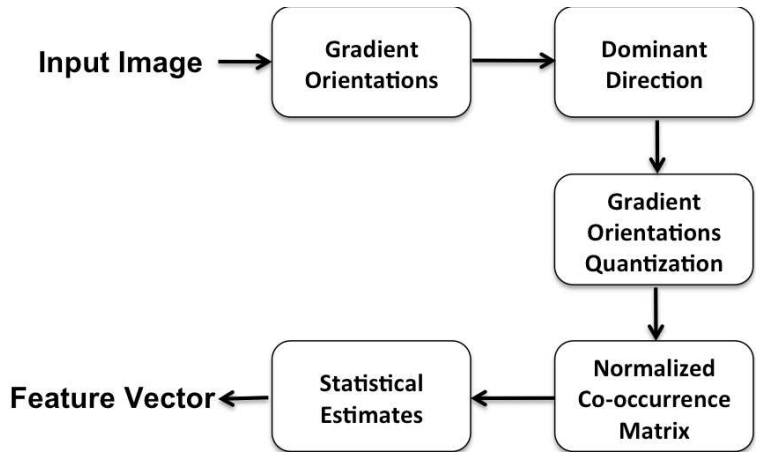


Fig. 2.4. GOSDM Block Diagram.

- Normalized co-occurrence matrix estimation
- Statistical estimates

Figure 2.4 shows the main steps of this approach. First, the image gradients of both vertical (I_v) and horizontal (I_h) directions are estimated using the method proposed by Farid and Simoncelli [122]. They described a set of optimized finite-size linear-phase separable kernels for differentiation of discrete multi-dimensional signals. In our implementation we used a 5-tap optimal coefficient differentiation filter.

Given the grayscale version of a texture image I, the directional derivatives, I_h and I_v in the horizontal and vertical direction respectively can be computed as follows:

$$I_h(m,n) = \sum_r \sum_s I(r,s) d_1(m-r) p(n-s)$$

$$I_v(m,n) = \sum_r \sum_s I(r,s) p(m-r) d_1(n-s)$$

with $d_1(.)$ being the L-tap coefficients of the differentiation filter and $p(.)$ being the interpolator function. Farid and Simoncelli shown that a 5-tap optimal 1st order differentiator; $p(.)$ and $d_1(.)$ can be represented by the following coefficients [122]:

$$p = [0.037659\ 0.249153\ 0.426375\ 0.249153\ 0.037659]$$

$$d_1 = [0.109604\ 0.276691\ 0.000000\ -0.276691\ -0.109604]$$

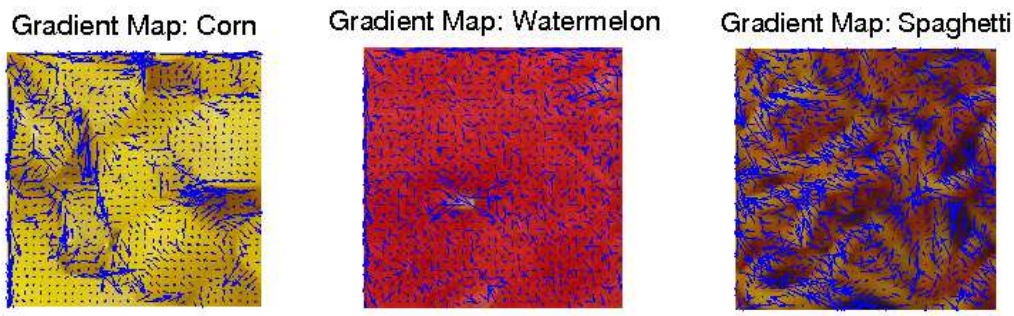


Fig. 2.5. Three Examples Of Gradient Orientation Fields. (a) Corn Texture, (b) Watermelon Texture, And (c) Spaghetti Texture. Note That Color Information Has Been Included For Illustration Purposes. For The Gradient Estimation The Image Is First Transformed To Gray Scale Space.

Figure 2.5 shows three examples of gradient orientation fields estimated using the above method.

Robustness against rotation is an important element to achieve good performance in texture categorization. Samples of the same texture may have different orientation values depending on camera viewpoint. The texture descriptor has to account for such orientation changes. In order to achieve robustness against rotation, the dominant orientation in the texture pattern is estimated. Based on [123], the horizontal and vertical derivative responses, (I_h, I_v), are represented in a 2-D space along the x and y axis respectively. Using a sliding window, the strength of both horizontal and vertical gradient for all the pixels inside the window are summed forming a single orientation vector. The final dominant direction is estimated by using the largest orientation vector. In our experiments, a 45° sliding window was used (Figure 2.6). In order to avoid noise effects all gradient responses $(\sqrt{(I_h(m,n)^2 + I_v(m,n)^2)})$ smaller than a threshold were not used for the dominant orientation estimation.

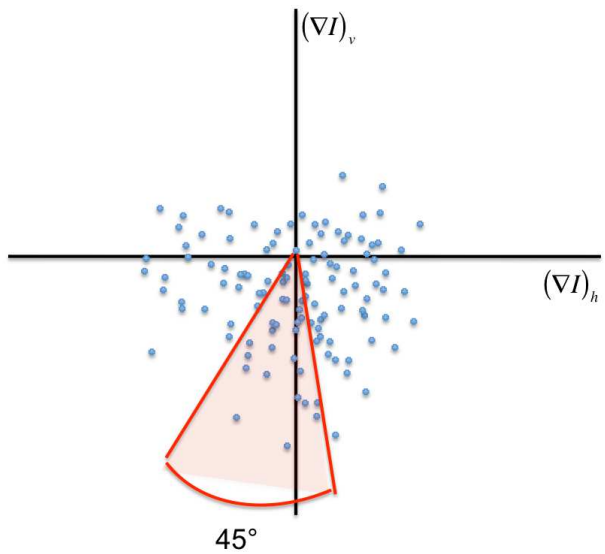


Fig. 2.6. Dominant Orientation Estimation Method To Achieve Robustness Against Rotation.

GOSDM describe the spatial relationship between gradient orientations by means of the occurrence of pairs of gradient orientation values. A quantization step is used to obtain a more compact representation of the oriented gradients:

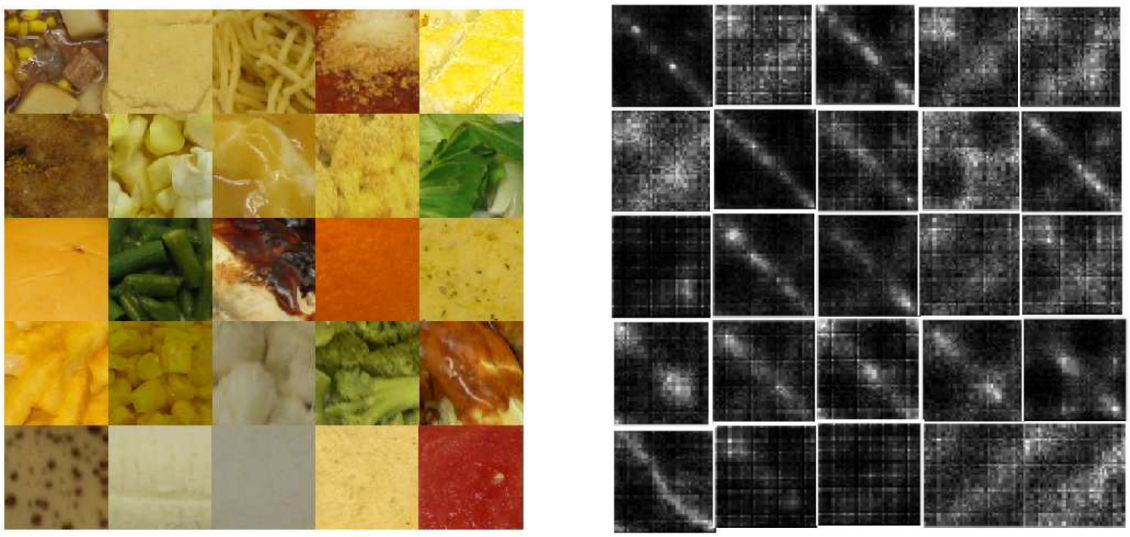
$$\theta_Q(m,n) = Q(\theta(m,n)) = (k+1/2)\Delta_Q + \theta_{min}$$

$$with\ \theta(m,n) \in [\theta_{min} + k\Delta_Q, \theta_{min} + (k+1)\Delta_Q] \tag{2.12}$$

$$for\ k = 0, ..., \Theta - 1.$$

where $\theta_Q(m,n)$ is the quantized gradient orientation at location (m,n), $\theta(m,n)$ is the gradient orientation with respect to the dominant orientation, $\Delta_Q = \frac{\theta_{max}-\theta_{min}}{\Theta}$, and Θ is the number of quantization levels.

Let $\mathbf{P_d}$ be the $\Theta \times \Theta$ co-occurrence matrix for displacement vector $\mathbf{d} = (d_x, d_y)$. Then, the entry (i,j) of $\mathbf{P_d}$ represents the number of occurrences of the pair of



(a) Original Textures (b) GOSDM

Fig. 2.7. Examples Of GOSDMs For Various Textures. a) Original Textures. b) Corresponding GOSDM With $d = (4, 0)$. (All images are converted to gray scale in our classification scheme).

quantized gradient orientations $\theta_Q(m, n) = i$ and $\theta_Q(t, v) = j$ that are $\mathbf{d}$ apart. Formally, this is given by Equation 2.13

$$P_{\mathbf{d}}(i, j) = |\{\{(m, n), (t, v)\} : \theta_Q(m, n) = i, \theta_Q(t, v) = j\}| \tag{2.13}$$

where $|\cdot|$ is the cardinality of the set, $(t, v) = (m + d_x, n + d_y)$, and $\theta_Q(., .)$ is the quantized version of the oriented gradient matrix. A final normalization step is done to obtain the probability-based version of the co-occurrence matrix. Thus, we can define the normalized oriented gradient co-occurrence matrix for an image with R pixels, $\Gamma_{\mathbf{d}}$, as $\Gamma_{\mathbf{d}} : \Theta \times \Theta \rightarrow [0, 1]$ where:

$$\Gamma_{\mathbf{d}}(i, j) = \frac{P_{\mathbf{d}}(i, j)}{R} \tag{2.14}$$

Figure 2.7 shows several examples of GOSDM for various food textures.

Scale information is an important component of texture characterization. Often highly discriminative structural elements of food textures are found at a particular scale. This effect can be seen in Figure 2.8, where small and large scale details are

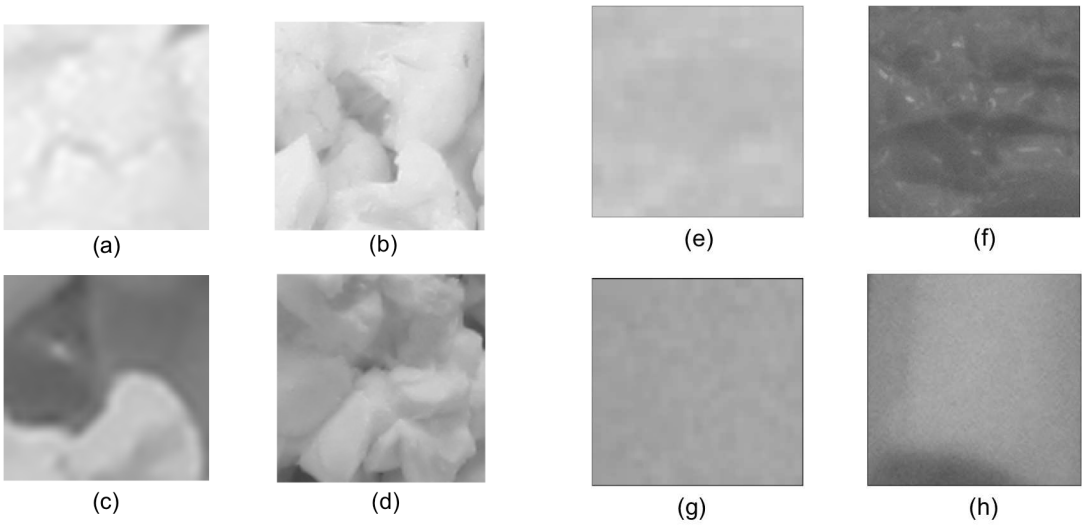


Fig. 2.8. Comparison Of Four Food Textures At Two Scale Levels. (a) Small Scale Cauliflower, (b) Large Scale Cauliflower, (c) Small Scale Popcorn, (d) Large Scale Popcorn, (e) Small Scale Mayo, (f) Large Scale Mayo, (g) Small Scale String Cheese, (h) Large Scale String Cheese.

shown for various foods. In the first case for Figures 2.8.a, 2.8.b, 2.8.c, and 2.8.d the smaller scales, Figures 2.8.a and 2.8.c, provide more descriptive information about the texture than the larger scale, Figures 2.8.b and 2.8.d. In our preliminary experiments, samples shown in Figures 2.8.a and 2.8.c were considered as different texture classes, where Figures 2.8.b and 2.8.d were incorrectly labeled as the same class. In contrast, in the second case (Figures 2.8.e, 2.8.f, 2.8.g, and 2.8.h) the effect is reversed; the larger scale, examples shown in Figures 2.8.f and 2.8.h, contain more discrimination power than the smaller scale, Figures 2.8.e and 2.8.g. In this case Figures 2.8.f and 2.8.h were correctly classified, whereas the samples shown in Figures 2.8.e and 2.8.g were assigned to the same class.

In order to incorporate the information contained at different scales several magnitudes of the vector $\mathbf{d}$ (offset) have been investigated. Ranging from a very local neighborhood of pixels, *i.e.*, magnitude $\mathbf{d}$ equal to $2^0, 2^2, 2^4, ...$, up to more global scales, *i.e.* magnitude of $\mathbf{d}$ equal to $min(H/2, V/2)$. For each level we determined four GOSDMs

based on the following unitary angular direction vectors: $(1,0),(\sqrt{2}/2,\sqrt{2}/2),(0,1)$, $(-\sqrt{2}/2,\sqrt{2}/2)$. Note that this strategy does not guarantee scale invariance, it only captures information at different pixel distances in order to account for variations between texture patterns that are only found at a particular scale. Scale invariance it is only achieved when the GOSDM feature values are identical for different scales or resolutions of the object. In our texture experiments in Chapter 4, we used three different given offsets $d = 1, 4, 16$ and angular orientations $(1,0),(\sqrt{2}/2,\sqrt{2}/2),(0,1),(-\sqrt{2}/2,\sqrt{2}/2)$ in order to estimate the GOSDMs.

To compress the large amount of information in GOSDM, while preserving their relevance, several features have been estimated. A subset of the proposed features from the GLCMs by Haralick [117] have been estimated from each GOSDM. These are *Correlation (COR), Angular Second Moment (ASM), Entropy (ENT), Contrast (CON), and Homogeneity (HOM).*

1. Correlation: The correlation measures the gradient orientation linear-dependence in the texture pattern. It is defined as:

$$COR = \sum_{i,j=0}^{N-1} \Gamma_{\mathbf{d}}(i,j) \left[\frac{(i-\mu_i)(j-\mu_j)}{\sqrt{\sigma_i^2 \cdot \sigma_j^2}} \right] \tag{2.15}$$

   where μ and σ are mean and variance, respectively. $\mu_i = \sum_{i,j=0}^{N-1} i\Gamma_{\mathbf{d}}(i,j)$, $\mu_j = \sum_{i,j=0}^{N-1} j\Gamma_{\mathbf{d}}(i,j)$, $\sigma_i^2 = \sum_{i,j=0}^{N-1}(i-\mu_i)^2\Gamma_{\mathbf{d}}(i,j)$, and $\sigma_j^2 = \sum_{i,j=0}^{N-1}(j-\mu_j)^2\Gamma_{\mathbf{d}}(i,j)$

2. Angular Second Moment: It describes the gradient orientation uniformity and homogeneity. ASM uses each $\Gamma_{\mathbf{d}}(i,j)$ as a weight for itself.

$$ASM = \sum_{i,j=0}^{N-1} \Gamma_{\mathbf{d}}(i,j)^2 \tag{2.16}$$

   The square root of the ASM, know as Energy, is sometimes used as a texture measure.

3. Entropy: this feature can be interpreted as the apparent randomness in the GOSDM.

$$ENT = \sum_{i,j=0}^{N-1} \Gamma_{\mathbf{d}}(i,j)(-\log \Gamma_{\mathbf{d}}(i,j)) \tag{2.17}$$

4. Contrast: this is also known as the "sum of square variances" [117]. It measures the amount of local variation of the gradient orientations, a large contrast value is due to large off-diagonal elements in the GOSDM. The weights increase exponentially as $(i-j)$ increases.

$$CON = \sum_{i,j=0}^{N-1} \Gamma_{\mathbf{d}}(i,j)(i-j)^2 \tag{2.18}$$

5. Homogeneity: measures visual uniformity.

$$HOM = \sum_{i,j=0}^{N-1} \Gamma_{\mathbf{d}}(i,j)/(1+|i-j|) \tag{2.19}$$

Once these measures are determined the final step is to construct the feature vector that describes the entire texture region. As mentioned above we used four GOSDM for each of the offset magnitudes corresponding to four angular directions $0\,^\circ, 45\,^\circ, 90\,^\circ, 135\,^\circ$. Thus, at each offset magnitude, d_i, we estimate the 20-dimensional feature vector f_{d_i} as: $f_{d_i} = [COR_{0\circ}\, ASM_{0\circ}\, ENT_{0\circ}\, CON_{0\circ}\, HOM_{0\circ}\, COR_{45\circ}\, ASM_{45\circ}\, ENT_{45\circ}\, CON_{45\circ}\, HOM_{45\circ}\, COR_{90\circ}\, ASM_{90\circ}\, ENT_{90\circ}\, CON_{90\circ}\, HOM_{90\circ}\, COR_{135\circ}\, ASM_{135\circ}\, ENT_{135\circ}\, CON_{135\circ}\, HOM_{135\circ}]_i$. Finally, for a $H \times V$ texture region, we concatenate the vector f_{d_i} for all levels $i = 2^0, 2^2, 2^4, ..., min(H/2, V/2)$ to obtain the final feature vector for GOSDM as:

$$\mathbf{f} = [f_{d_1}\ f_{d_4}\ f_{d_{16}}\ ...\ f_{d_{min(H/2,V/2)}}] \tag{2.20}$$

Entropy-Based Categorization and Fractal Dimension Estimation (EFD)

Our second texture feature is based on fractal information. Pentland showed how fractal functions can be used for modeling 3-D natural surface shapes and are

widely used in graphics for generating natural-looking shapes [107]. The fundamental property of a fractal is that it has a fractal dimension (FD). Richardson shown that the statistics of a scalloped curve are invariant with respect to transformations of scale [124]. "If one examined a coastline, he/she would see a scalloped curve formed by bays and peninsulas. If one examined further a finer-scale map of the same region, he/she would again see the same type of curve. If one wanted to measure the length of coastline, he/she would take a measuring instrument of size ϵ, determining that n such instruments would "cover" the curve or area to be measured" [124]. The length could be measured by:

$$M = n\epsilon^D \tag{2.21}$$

where M is the metric property to be measured (length), and D is the topological dimension of the measuring instrument. The question becomes, how would someone measure the length of the curves that are smaller than the size of the measuring tool? Mandelbrot [125] proved that in order to obtain consistent measurements in such scenarios, the notion of a fractional dimension had to be introduced. The FD is the only consistent fractional power that provides the correct adjustment factor for all those details smaller than ϵ.

Pentland showed that images are "fractal surfaces" when they can be approximated by a single fractal function over a finite range of scales [107]. In particular, he showed that we can model image surfaces as *Fractal Brownian Functions*). Although, images of natural scene can be described by fractal information, FD alone does not provide high discrimination since textures may have the same FD due to combined differences in directionality and coarseness [125]. These uncertainties can be addressed by multifractal analysis [126, 127], where a point categorization is defined on the object function based on some criteria. The FD is estimated for every point set according to this categorization.

Let Υ be a finite Borel regular measure on R^2. For $x \in R^2$, denote $B(x, \epsilon)$ as the closed disc with center x and radius $\epsilon > 0$. $\Upsilon(x, \epsilon)$ can be described by an exponential

function of ϵ, i.e. $\Upsilon(x,\epsilon) = n\epsilon^D$, same as (Equation 2.21). Then the *local density function* of x is defined as:

$$D(x) = \lim_{\epsilon \to 0} \frac{\log \Upsilon(x,\epsilon)}{-\log \epsilon} \tag{2.22}$$

The $D(x)$ describes how the measurement Υ satisfies the power law behavior (Equation 2.21).

For any $\alpha \in R$, define:

$$\Sigma_\sigma = \{x \in R^2 \,:\, D(x) = \frac{\log(\Upsilon(x,\epsilon))}{-\log \epsilon} = \sigma\} \tag{2.23}$$

where Σ_σ, is the set of all image points x with local density σ. This set is irregular, and thus, is described by its fractal dimension $dim(\Sigma_\sigma)$. As a result, we can obtain a point categorization $\{\Sigma_\sigma \,:\, \sigma \in R\}$ of the image with a multifractal denoted as: $\{dim(\Sigma_\sigma) \,:\, \sigma \in R\}$. In [128], several categorization approaches were taken including a smoothed gaussian filtered version of the image, energy of the gradients, and the sum of laplacian. In this thesis we propose two categorization criteria: entropy (EFD) and Gabor filter banks (GFD). In Section 4.2 we show how entropy and Gabor filter banks result in better categorization criteria than energy of the gradients or sum of laplacian for texture representation.

Entropy can be seen as a measure of local signal complexity [96]. Regions of the image corresponding to high complexity (high level of detail) tend to have higher entropy. In general, complexity is independent of scale and position [129], hence we can categorize a texture by selecting areas with homogeneous entropy levels. This approach can be seen as an attempt to characterize the variation of roughness of homogeneous parts of the texture in terms of complexity. Also, the entropy function is a Borel measurable function [130], thus, multifractal analysis can be used. Once the entropy is estimated for pixels in the texture image (Equation 2.10), we cluster regions where the entropy function exhibits similar values. For a given entropy υ level, Υ_υ represents the set of pixels whose values are 'δ' close to υ, *i.e.* $\{x : x \in H \times V \, and \, H_x \in (\upsilon, \upsilon + \delta)\}$, for some arbitrary δ. Once this pixel categorization is completed, we estimate $dim(\Upsilon_\upsilon)$, the FD for each Υ_υ.

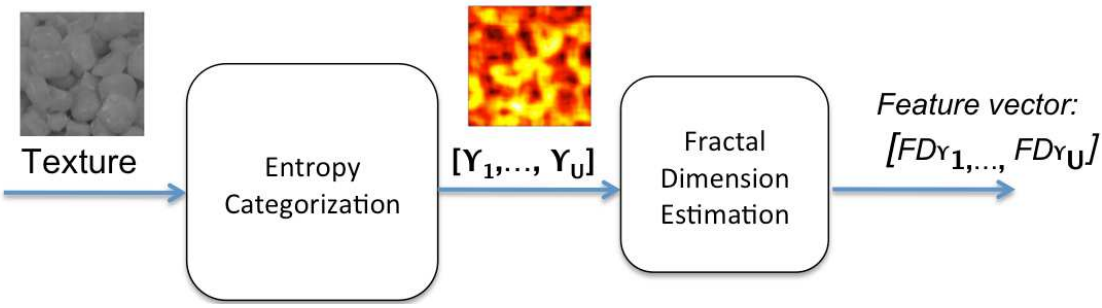


Fig. 2.9. Entropy-Based Multifractal Analysis Block Diagram (EFD).

In practice, the 3D space can be divided into a mesh of cubical boxes of size ϵ, for a total of $\epsilon \times \epsilon \times \epsilon$ boxes. The image is represented in the 3D space by the triple formed by its gray scale pixel values, and x and y pixel coordinates. N_ϵ is the number of boxes that intersect with the surface of the image in order to cover the set of pixels Υ_v. This approach is known as *Box-counting* [131]. Instead of using linear fitting in the space N_ϵ vs. $1/\epsilon$, the FD is estimated as the weighted sum of ratios $\frac{log(N_\epsilon)}{1/\epsilon}$ [119]. As a result of this process, the FD is better approximated for small pixel sets. In our experiments (Section 4.2) we used 30 different values of ϵ ranging from 1 to 30 pixels. We uniformly partitioned the estimated entropy of the texture image into 4 levels. The total dimension of our EFD feature vector is equal to 120.

For each of the entropy levels we estimate the FD_{Υ_v}. The final texture feature is formed by fusing all the FD_{Υ_v} into one feature vector, $\mathbf{f}$. Figure 2.9 summarizes the proposed approach.

Gabor-Based Image Decomposition and Fractal Dimension Estimation (GFD)

One of the challenges in texture modeling is incorporating descriptions of both the geometries of the light sources and imaging systems and the optical properties of the surface materials. Bovik *et. al.* modeled textures as "irradiance patterns" which are distinguished by a high concentration of localized spatial frequencies [111] using Gabor filters.

2-D Gabor filters can represent images as multiple complex-modulated subimages. The energy in each subimage is located in both spatial frequency and in space allowing accurate localization of boundaries occurring between textures with different dominant characteristics. In this section we review the basic properties of the Gabor channel filters, and introduce our third texture descriptor combining Gabor filters and multifractal analysis theory.

A Gabor impulse response in the spatial domain consists of a sinusoidal plane wave of specific orientation and spatial frequency, modulated by a two-dimensional Gaussian envelope. This is given by:

$$g(x,y) = exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)\right]\cos\left(2\pi Ux + \varphi\right) \tag{2.24}$$

where U is the modulation frequency, x, y are coordinates in the spatial domain, and σ_x and σ_y are the standard deviations in the x and y direction. The 2D Fourier Transform (FT) is:

$$G(u,v) = exp\left\{-\frac{1}{2}\left[\frac{(u-W)^2}{\sigma_u^2} + \frac{v^2}{\sigma_v^2}\right]\right\} \tag{2.25}$$

where $\sigma_u = \frac{1}{2\pi\sigma_x}$, $\sigma_u = \frac{1}{2\pi\sigma_y}$, and u and v are coordinates in the frequency domain. This filter can be seen as a local band-pass filter with optimal joint localization properties in both the spatial domain and the spatial frequency domain.

A Gabor filter-bank consists of a set of Gabor filters with Gaussians of different sizes modulated by sinusoidal plane waves of different orientations from the same mother Gabor filter as defined in Equation 2.24. A Gabor filter-bank can be represented as:

$$g_{m,n}(x,y) = a^{-m}g(\tilde{x},\tilde{y}), \quad a > 1 \tag{2.26}$$

where $\tilde{x} = a^{-m}(x\cos\theta + y\sin\theta)$, $\tilde{y} = a^{-m}(-x\sin\theta + y\cos\theta)$, $\theta = n\pi/K$ (K = total orientation, and $n = 0, 1, ..., K-1$), and $g(\cdot,\cdot)$ is defined in Equation (2.24). An input image $I(x,y), x, y \in \Omega$, being Ω -the set of image points, is convolved with a two-dimensional Gabor function $g(x,y), x, y \in \Omega$ to obtain a Gabor filtered image as follows:

$$I_{g_{m,n}}(x,y) = \iint_{\Omega} I(\xi,\eta)g(x-\xi, y-\eta)\, d\xi\, d\eta \tag{2.27}$$

Equivalently, the discrete Gabor filtered output is given by a 2D convolution:

$$I_{g_{m,n}}(r,c) = \sum_r \sum_s I_E(m-r, n-s) {g_{m,n}}^*(r,s),\ m = 0,1,...,S-1, n = 0,...,K-1 \tag{2.28}$$

where * indicates the complex conjugate.

The non-orthogonality of Gabor filters implies that there is redundant information in the filtered image. The following design described in [118] guarantees the adjacent half-peak contours touch each other, after choosing the number of orientations K, the number of scales S, and the upper and lower center frequencies U_h and U_l:

$$\begin{aligned} &a = \left(\frac{U_h}{U_l}\right)^{\frac{1}{S-1}},\ \sigma_u = \frac{(a-1)U_h}{(a+1)\sqrt{2\ln 2}} \\ &\sigma_v = \tan\left(\frac{\pi}{2K}\right)\sqrt{\frac{U_h^2}{2\ln 2} - \sigma_u^2},\ W = U_h \\ &m = 0,1,\cdots,S-1, \quad n = 0,1,\cdots,K-1 \end{aligned}$$

Features related to the local spectrum have been proposed in the literature and used for texture classification and/or segmentation [112, 118, 132–134]. The most popular ones include the 1st and 2nd moment statistics, μ_{mn} and σ_{mn}, defined as:

$$\begin{aligned} \mu_{mn} &= \frac{\sum_r \sum_c |I_{g_{m,n}}(r,c)|}{H \times W} \\ \sigma_{mn} &= \frac{\sqrt{\sum_r \sum_c (|I_{g_{m,n}}(r,c)| - \mu_{mn})^2}}{H \times W} \end{aligned} \tag{2.29}$$

The final feature vector of size $S \times K$ is defined as:

$$\mathbf{f} = [\mu_{00} \quad \sigma_{00} \quad \cdots \quad \mu_{(S-1)(K-1)} \quad \sigma_{(S-1)(K-1)}] \tag{2.30}$$

We propose the use of Gabor filterbanks as a categorization criteria for multifractal analysis. The idea is to use multifractal theory and a series of image decompositions using basis functions of varying spatial frequency. Note that in order to provide rotation robustness to our descriptor we use each of the Gabor filters with respect to the dominant orientation of the texture pattern. The dominant orientation was estimated as shown in Figure 2.6.

For each scale and orientation, we estimate the fractal dimension of the $I_{g_{m,n}}(r,c)$, $FD_{I_{g_{m,n}}}$. The final descriptor becomes:

$$\mathbf{f} = [FD_{I_{g_{1,1}}}\ FD_{I_{g_{1,2}}}\ ...\ FD_{I_{g_{S,K}}}] \tag{2.31}$$

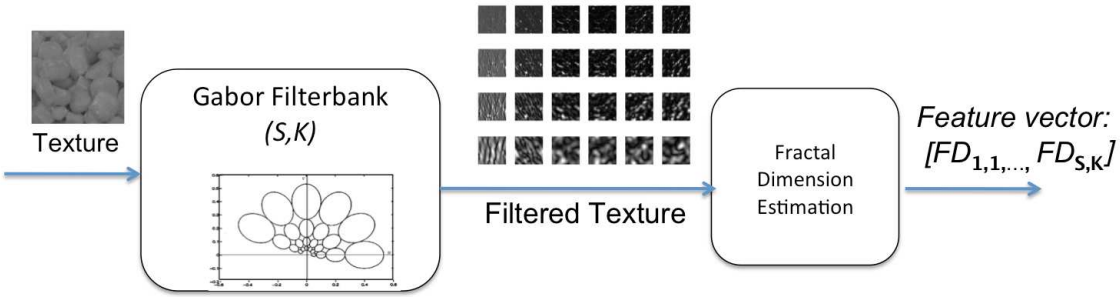


Fig. 2.10. Gabor-Based Multifractal Analysis Block Diagram (GFD).

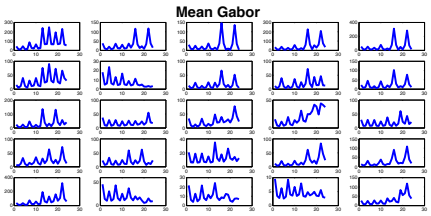
In our experiments (Section 4.2) we used $S = 4$, $K = 6$, and we examined 5 values of ϵ (size of boxes in the *counting box* method) ranging from 1 to 5 pixels. This resulted in a 120 dimensional feature vector.

Figure 2.10 illustrates the multifractal analysis using Gabor filterbanks. Figure 2.11 shows an example of the two different Gabor signatures (2.30) and (2.31) for the sample texture mosaic. In Section 4.2 the evaluation of the proposed texture descriptor is discussed.

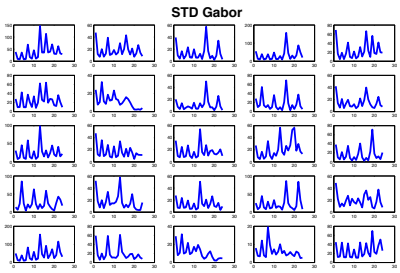
2.3 Local Features

An object can also be characterized by its composition of local image regions. A common way to capture local image information is by low-level local features. High-dimensional low-level features are grouped together using a codebook model forming a visual word vocabulary. In object classification, each object can be represented using the visual vocabulary [85, 135–137]. In this section, we describe the low-level features we have investigated. All our local features are obtained from local neighborhoods around points of interest detected in the object or segmented region. We investigated two point detectors (Section 2.3.1) namely *Difference of Gaussians* (DoG) [138] and *Entropy*-based point detector, and several local descriptors (Section 2.3.2) including *SIFT* descriptor, *Red-SIFT* descriptor, *Green-SIFT* descriptor, *Blue-SIFT* descriptor, *SURF* descriptor, *TAMURA* local descriptor, *Steerable filters*, *DAISY* descriptor,

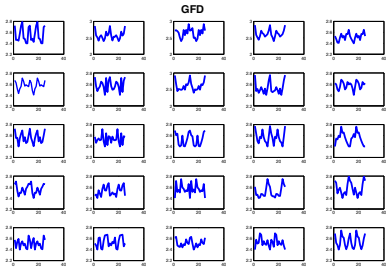
(a) Original Textures



(b) Mean (2.30)



(c) Standard Deviation (2.30)



(d) GFD (2.31)

Fig. 2.11. Examples Of Gabor Features. (a) Original Textures. (b) Mean Of The Energy. (c) Standard Deviation Of The Energy. (d) GFD Signature.

Gabor local descriptor, *local color statistics*, *sum of MPEG-7 edge descriptors*, and *local-GOSDM*.

2.3.1 Feature Detection and Points of Interest

Local features are extracted around points located in regions of the image. These locations are often called *points of interest*. They are described by the appearance of the group of neighboring pixels surrounding the point location. The idea is to find points in the object where we can reliably find them in other samples of the same object, and the features around such points of interest are self-similar for different illumination and viewpoints changes.

Forstner [139], and Harris and Stephens [140] were the first to propose using local maxima of rotationally invariant scalar measures, derived from the auto-correlation matrix, to locate points of interest. Using a Taylor series expansion of the image function $I(\mathbf{x_i} + \Delta\mathbf{u}) \approx I(\mathbf{x_i}) + \nabla I(\mathbf{x_i})$, we can approximate the auto-correlation surface as:

$$R = \sum \omega(\mathbf{x_i})[I(\mathbf{x_i} + \Delta\mathbf{u}) - I(x_i)]^2 = (\Delta\mathbf{u})^T A \Delta\mathbf{u} \tag{2.32}$$

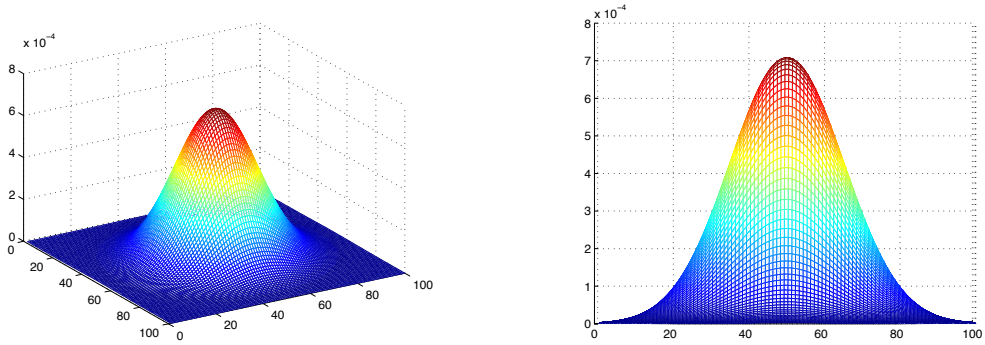
where A is known as the *structure tensor* and can be written as:

$$A = \omega * \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \tag{2.33}$$

where I_x and I_y are the directional derivatives with respect to the displacement vector $\mathbf{u}$, $\omega = \sigma^2 g(\sigma_I)$ is a weighting kernel, with $g(\sigma_I)$ a bi-dimensional Gaussian function defined as:

$$g(\sigma_I) = \frac{1}{2\pi\sigma_I^2} exp^{-\frac{x^2+y^2}{\sigma_I}} \tag{2.34}$$

In general, scale invariance is a fundamental property in order to guarantee successful local description. Features have to show similar values at several scale representations of the object. One approach is to obtain local features at a variety of scales by performing the same operations at multiple resolutions in a pyramid and



(a) Uniform Gaussian Filter

(b) Uniform Gaussian Filter 1-D

Fig. 2.12. Example Of Gaussian Filter.

then matching features at the same level. However, in general the scale of the object is unknown and pyramid representation becomes infeasible. Also finding local features at many different scales and then matching them it is not computationally efficient. Instead, extracting features that are stable in both location and scale is a better solution [138, 141]. Early investigations into scale selection were performed by Lindeberg [83], who proposed using extrema in the Laplacian of Gaussian (LoG) function to find stable point locations.

Numerous studies have shown that the Gaussian kernel is optimal for computing the scale-space representation of an image [83,84,142]. The scale-space representation is a set of images represented at discrete levels of resolution (scales). Koenderink in [82] was the first to show that the scale-space must satisfy the diffusion equation (Equation 2.37) for which the solution is a convolution with the Gaussian kernel. Others such as Babaud [142], Lindeberg [83] and Florack [84] later confirmed the uniqueness of the Gaussian kernel for multi-scale representation.

In multi-scale representation, levels of the scale-space representation are created by the convolution with the Gaussian kernel (Figure 2.12):

$$L = L(x, y, \sigma_D) = g(\sigma_D) * I(x, y) \tag{2.35}$$

where I is the image and (x, y) the point location. The kernel is circularly symmetric and parameterized by one scale factor σ_D. Thus, the structure tensor (Equation 2.33) can be rewritten as:

$$A = \sigma_D^2 g(\sigma_I) * \begin{bmatrix} L_x^2(x, y, \sigma_D) & L_x(x, y, \sigma_D)L_y(x, y, \sigma_D) \\ L_x(x, y, \sigma_D)L_y(x, y, \sigma_D) & L_y^2(x, y, \sigma_D) \end{bmatrix} \tag{2.36}$$

The next step is to select the scale that guarantees feature stability (similar feature values for different scales). The idea is to select the characteristic scale, for which a given function attains an extremum over scales. Some examples of these functions include:

- Laplacian of Gaussians: $LoG = \sigma_I |L_{xx}(x, y, \sigma_D) + L_{yy}(x, y, \sigma_D)|$
- Harris function: $det(\mu(x, y, \sigma_I, \sigma_D)) - \alpha trace^2(\mu(x, y, \sigma_I, \sigma_D))$
- Difference of Gaussians: $DoG = |I(x, y) * g(\sigma_I) - I(x, y) * g(k\sigma_I)|$
- Hessian trace and determinant: $Max(|trace(H)|)$ and $max(|det(H)|)$. With H the Hessian matrix.

In all cases these measures are only obtained from the grayscale version of the image since Gaussian filtering introduces new chromaticities that can decrease the efficiency of the point of interest. In this work we have investigated two point detectors based on multi-scale representation: Difference of Gaussians (DoG) and an entropy-based Point Detector.

Difference of Gaussians (DoG)

DoG has proven to be efficient to detect stable points of interest in scale-space representation [138]. One of the reasons for choosing this function to estimate the points of interest is that as the smoothed images, L, need to be estimated in any case for scale-space feature description, then DoG can therefore be estimated by applying

a simple subtraction operation. DoG is an approximation of Laplacian of Gaussians (LoG), $\sigma^2\nabla^2 g(\sigma_D)$. This can also be understood from the heat diffusion equation:

$$\frac{\partial g}{\partial \sigma} = \sigma \nabla g \tag{2.37}$$

where the diffusion equation can be approximated by:

$$\sigma \nabla g = \frac{\partial g}{\partial \sigma} \approx \frac{g(x, y, k\sigma) - g(x, y, \sigma)}{k\sigma - \sigma} = \frac{g(x, y, k\sigma) - g(x, y, \sigma)}{\sigma(k-1)} \tag{2.38}$$

This shows that when the DoG function has scales differing by a constant factor

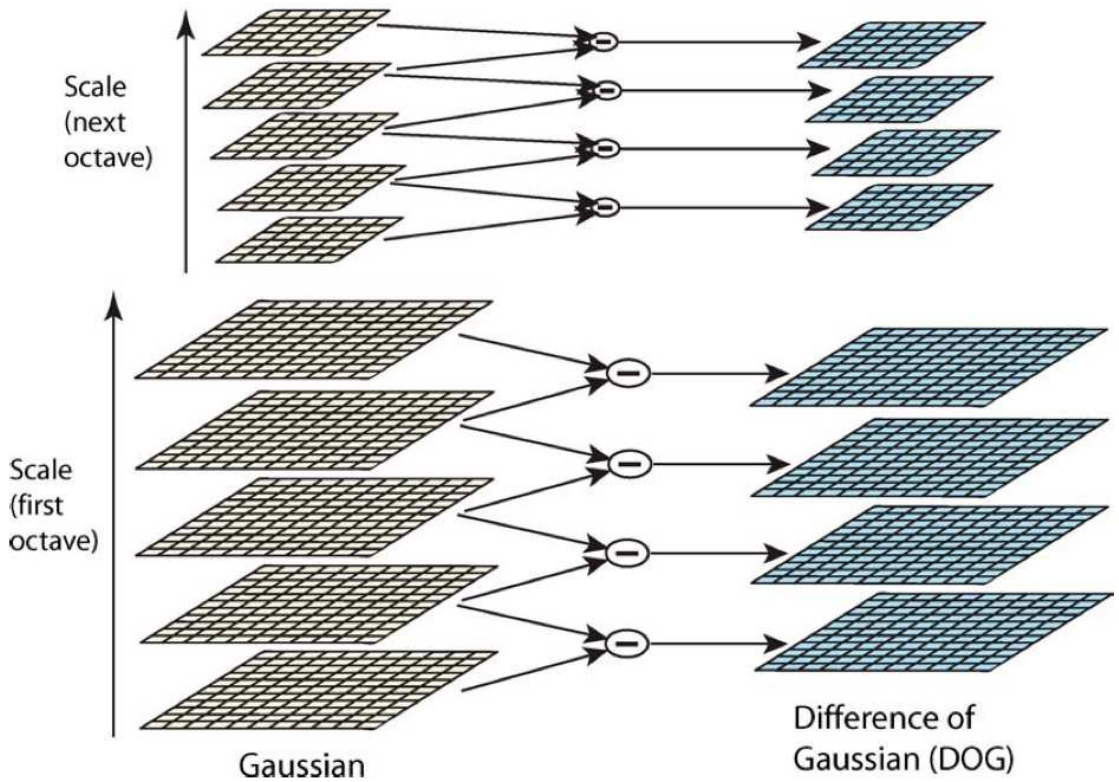


Fig. 2.13. Difference Of Gaussian Formation [138].

it incorporates the σ^2 scale normalization required for the scale-invariant Laplacian [138].

An efficient approach to construct the DoG is shown in Figure 2.13, where the original image, I, is convolved with Gaussian kernels to produce smoothed images,

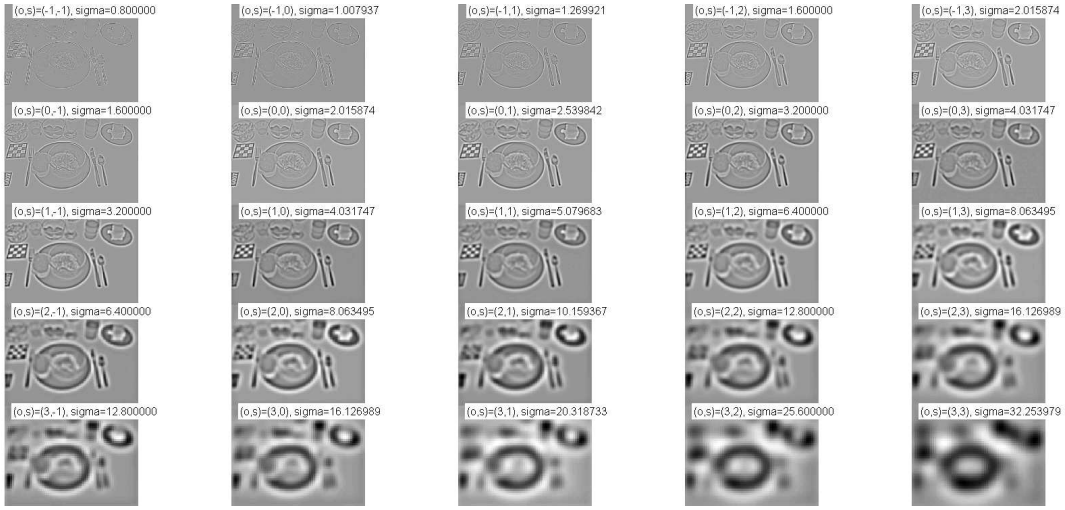


Fig. 2.14. Example Of Difference Of Gaussians To Locate Points Of Interest.

L, separated by a constant factor k in scale-space, as shown in the left column of Figure 2.13. After this, the next step is to subtract adjacent image scales to produce the actual difference-of-Gaussian images. Figure 2.14 shows several examples of difference-of-Gaussian images for a given eating occasion image. Once a complete octave has been processed, the Gaussian image is resampled to twice the initial value of σ by taking every second pixel in each row and column [138]. The final step is to detect the local extrema (maxima and minima) of $DoG(x, y, \sigma)$. This is accomplished by comparing each sample point to its eight neighbors in the current image and nine neighbors in the scale above and below. The sample point is selected only if it is larger (maxima) than all of his neighbors or smaller (minima) than all of them.

The DoG function has a strong response along edges of the objects. These responses describe large principal curvatures across the edge but small ones in the perpendicular direction to the edge. Harris and Stephens proposed a method to detect and distinguish corner points from edges based on the ratio between the eigenvalues, λ_1, λ_2, of the structure tensor matrix, $\mathbf{A}$, (Equation 2.36) [140]. Edge points are detected by examining the following measure:

$$\frac{Tr(\mathbf{A})^2}{Det(\mathbf{A})} < \frac{(r+1)^2}{r} \tag{2.39}$$

where r is the ratio between the largest (λ_1) and smallest (λ_2) eigenvalue of the structure tensor. Points that satisfy the above expression are considered edge responses, and thus, they are eliminated from the candidate list of points of interest. In general, the large success of the DoG point detector, and the other point detectors based on Gaussian scale-space theory is based on the repeatability of the point over a large range of scales.

Entropy-Based Point Detector

Kadir [129] questioned the criteria of finding self-similar points at stable scale-space representations because it does not necessarily detect points located at the most salient areas. Features that exist over large ranges of scale may exhibit self-similarity which Kadir claimed are regarded as non-salient. Gilles defined saliency in terms of local signal complexity or unpredictability. He measured local signal complexity to detect salient areas by using Shannon entropy (Equation 2.10) [143].

In our work, a salient point detector robust against scale variation was estimated by choosing scales at which the entropy is maximum. Given a smoothed version of the image at a scale σ, $L(x, y, \sigma)$, for each pixel, the entropy, H was estimated using Equation 2.10 in a neighborhood of size $round(16\sigma)$, where σ represents the scale of the smoothed image, $L(x, y, \sigma)$. Then the set of scales, S at where the entropy peaked was obtained by:

$$S = \{s : \frac{\partial^2 H(x, y, \sigma)}{\partial \sigma^2} < 0\} \tag{2.40}$$

Connected-components was used on the scales belonging to S, and finally, non maxima suppression was used to obtain the final points of interest. Figure 2.15 shows an example of the entropy-based point detector.

2.3.2 Local Descriptors

After detecting the points of interest, we need to define a set of descriptors that capture visual information of a patch or neighborhood around the point of interest.

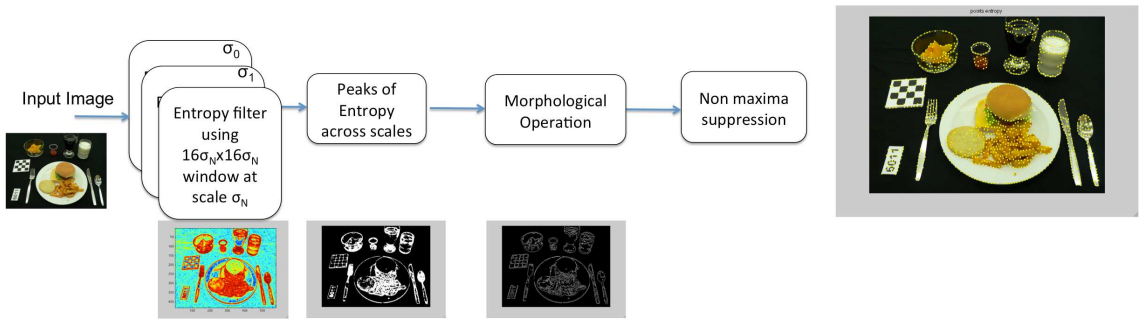


Fig. 2.15. Entropy-Based Point Detector.

This is known as a *local feature* or a *local descriptor*. Information content and invariance are two important properties of local descriptors. The objective is to find descriptors that maximize the information content of a local region of the object while having similar values in other samples of the same object. The information content is the quantity of information conveyed by the local descriptor. One of the main limitations of local features is the small size of the local neighborhoods. The repetitiveness rate of the point of interest becomes critical to guarantee the success of local features.

There are many description techniques that can be used to estimate the representation of a local feature. In this section, we overview local descriptors including the Scale Invariance Feature Transform (SIFT), Speeded-Up Robust Features (SURF), DAISY descriptor, as well as the other local features that we have also considered including steerable filters, Tamura filters, Gabor filters, MPEG-7 Edge Descriptors, and GOSDMs.

SIFT Descriptor

As discussed above, the point detector provides an image location, scale, and orientation to each point. By taking into account these elements we can create a local 2D coordinate system in which to describe the local image region providing invariance to scale and orientation. The next step is to estimate a descriptor for the local image

region that is highly distinctive, and at the same time it is as invariant as possible to remaining variations, such as change in illumination. SIFT features have proven to be very successful at achieving several types of invariance. SIFT descriptors are formed by estimating the gradient at each pixel in a 16×16 window around the detected points of interest, using the appropriate level of the DoG pyramid at which the point of interest was detected. The gradient magnitudes are weighted by a Gaussian fall-off function (illustrated as a blue circle in Figure 2.16) in order to reduce the influence of gradients far from the center or point of interest. Neighboring points located farther from the center are more affected by small mis-registrations [138]. In each 4×4 quadrant, a gradient orientation histogram is formed by adding the weighted gradient to one of eight orientation histogram bins. To reduce the effects of location and dominant orientation misestimation, each of the original 256 weighted gradient magnitudes is added to $2 \times 2 \times 2$ histogram bins resulting in a 128 non-negative values.

To further make the descriptor robust to other photometric variations, values are clipped to 0.2 and the resulting vector is once again renormalized to unit length. This results into the final 128 values of the SIFT descriptor [138].

We also investigated the integration of SIFT descriptors with color information. We obtained SIFT descriptors from the *R,G,B* color space components in the same fashion as proposed by Lowe. We refer to these descriptors as *Red-SIFT, Green-SIFT, and Blue-SIFT*.

SURF Descriptor

The second type of local features investigated was the Speeded-Up Robust Features (SURF). SURF features were proposed as a joint point detector and descriptor [123]. The point detector consists of constructing Hessian matrices using integral images to speed up the detection process. As a result of this, mostly blob-like structures are detected at locations where the determinant is maximum. In this thesis, only

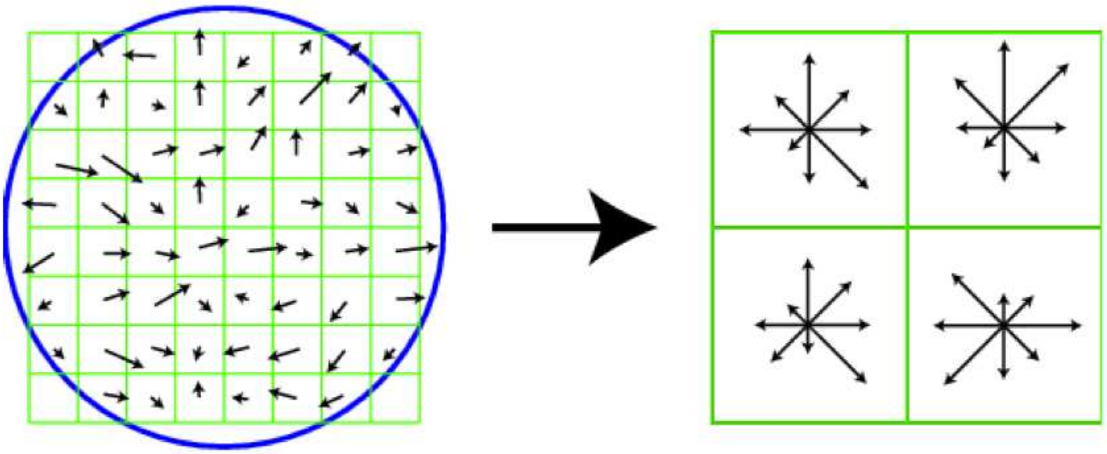
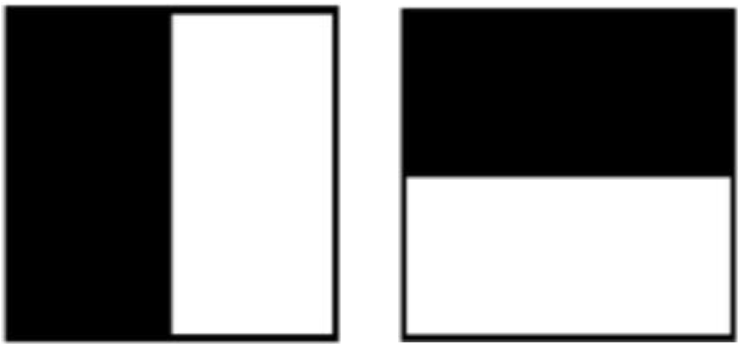


Fig. 2.16. SIFT Descriptor Representation [138]. (Left) Gradient Orientations And Magnitudes Are Estimated At Each Pixel Location And Weighted By A Gaussian Fall-off Function (Blue Circle). (Right) A Weighted Gradient Orientation Histogram Is Then Computed In Each Subneighborhood Using Interpolation. Note That This Figure Shows An 8×8 Pixel Patch And A 2×2 Descriptor Array, Lowe In His Experiments Used A 16×16 Pixel Patch And A 4×4 Array Of Eight Bin Histogram.

the description approach was investigated since SURF detectors failed at detecting points found at structures other than blob-like structures.



(a) Haar Horizontal Filter (b) Haar Vertical Filter

Fig. 2.17. Haar Filters. (a) Horizontal. (b) Vertical.

For the construction of the descriptor, the first step consists of estimating square regions centered around each point of interest. The size of this windows was set equal to 20×20. The Haar wavelet in the horizontal, d_x, and vertical, d_y, directions (Figure 2.17) are used to filter each subregion of the original neighborhood. Then, the wavelet responses d_x and d_y are accumulated over each sub-region and form the first set of values in the feature vector. The absolute values of the responses, $|d_x|$ and $|d_y|$, are also estimated for gradient robustness forming the second set of entries in the feature vector. As a result, each sub-region has a four-dimensional descriptor vector v for its underlying intensity structure $v = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|)$ [123]. The main strength of this descriptor is its speed. In [123] it was shown that SURF can be estimated in a very fast manner achieving performance close to other descriptors such as SIFT. However, SURF has two main disadvantages:

- SURF is not as invariant to illumination and viewpoint changes as SIFT.
- SURF introduces artifacts that degrade the matching performance when used with many points of interest.

Tamura Local Descriptor

Tamura *et al.* proposed 6 perceptual texture measures: coarseness, contrast, directionality, line-likeness, regularity, and roughness. These were selected by psychological experiments to better describe human perception [144]. Although originally Tamura proposed these measures as global texture descriptions of objects, we have investigated the use of Tamura features as local descriptors by building visual codebooks composed of the original Tamura features. In our experiments we have found that coarseness, contrast, and directionality are the most discriminative [72]. Also from Tamura's work in [144], it was indicated that these three measures correlate strongly with human perception.

Coarseness: coarseness provides information about the size of primitive texture elements [144]. The higher the coarseness, the rougher the texture. Coarseness esti-

mation starts by finding the average grayscale value at every point in neighborhoods whose sizes are powers of two:

$$A_k(x,y) = \sum_{r=x-2^{k-1}}^{x+2^{k-1}-1} \sum_{s=y-2^{k-1}}^{y+2^{k-1}-1} \frac{I(r,s)}{2^{2k}} \tag{2.41}$$

where $I(r,s)$ is the gray level value of the pixel at point (r,s). The second step is to estimate differences between the non overlapping neighborhoods on opposite sides for every point (x,y) in the horizontal and vertical directions:

$$E_k^h(x,y) = \|A_k(x+2^{k-1},y) - A_k(x-2^{k-1},y)\| \tag{2.42}$$

and

$$E_k^v(x,y) = \|A_k(x,y+2^{k-1}) - A_k(x,y-2^{k-1})\| \tag{2.43}$$

Then, at each point the size coresponding to the highest difference is selected, $S(x,y) = argmax_{k=1,\ldots,5}\{max_{d=h,v}(E_k^d(x,y))\}$. Finally, the average over 2^S is used as a measure of coarseness of the image:

$$f_{crs} = \frac{1}{HV} \sum_{x=1}^{H} \sum_{y=1}^{V} 2^S(x,y) \tag{2.44}$$

Contrast: contrast is influenced by the dynamic range of the gray levels, sharpness of edges and period of repeating patterns [144]. Tamura showed that contrast can be a measure of image quality and is influenced by four factors: dynamic range of the gray-levels, polarization of the distribution of black and white in the gray-level histogram, sharpness of edges, and the period of repeating patterns. For this reason he discarded the notion of contrast introduced by Haralick in [104] as: $contrast = \sum_n n^2 \sum\sum_{|i-j|=n} I(i,j)$. Instead, the contrast measure that Tamura proposed can be estimated as:

$$f_{con} = \frac{\sigma}{(\alpha)^n} \tag{2.45}$$

where σ is the standard deviation about the mean (μ) of the gray-level probability distribution, the α parameter is defined by $\alpha = \mu/\sigma$. Usually n can values 8, 4, 2, 1, 1/2, 1/4, or 1/8.

Directionality: directionality does not refer to orientation itself, but whether or not the texture has a particular orientation [144]. This means that two textures differing only in the orientation are considered to have the same directionality. The histogram of local edge probabilities relative their directional angle it is used. The orientation of the gradient is estimated as follows: $\theta = \pi/2 + \tan^{-1}(I_v/I_h)$. The histogram ($H_D(k)$) can be obtained by quantizing θ and counting the points with the magnitude $((|\Delta_h||\Delta_v|)/2)$ above a threshold. One way to measure directionality quantitatively from H_D is to compute the sharpness of the peaks. The approach which was adopted is to sum the second moments around each peak from valley to valley, if multiple peaks are determined to exist. This measure is defined as follows:

$$f_{dir} = 1 - rn_p \sum_{p}^{n_p} \sum_{\phi \in \omega_p} (\phi - \phi_p)^2 H_D(\phi) \tag{2.46}$$

where n_p is the number of peaks, ϕ_p is the pth peak position of H_D, ω_p is the range of the pth peak between valleys, r is the normalizing factor related to the quantizing levels of ϕ, and ϕ is the quantized direction code (cyclically in modulo 180°).

Our local Tamura descriptor estimates f_{crs}, f_{con}, and f_{dir} in a neighborhood of the point of interest of size 16×16 pixels.

Steerable Filters

Steerable filters are randomly oriented filters proposed by Freeman and Adelson in [145]. These filters are combinations of Gaussian filters that permit fast estimation of odd and even edge and corner-like features at all possible orientations. The first order directional derivative filter is an example of a steerable filter [145]:

$$\nabla_{\mathbf{u}} G = G_{\mathbf{u}} = uG_x + vG_y = u\frac{\partial G}{\partial x} + v\frac{\partial G}{\partial y} \tag{2.47}$$

$\mathbf{u}$ is an unitary vector that steers the Gaussian kernel at a give direction. Following the same approach we can define the steerable n^{th} order derivative filter as $\nabla_n = \nabla^1_{\mathbf{u}} \cdots \nabla^n_{\mathbf{u}} G_{\mathbf{u}}$. This structure allows to construct derivative filters to increasingly

greater directional selectivity. Furthermore, higher order steerable filters can respond to potentially more than a single edge orientation at a given location, and can respond to both "bar edges" (thin lines) and the classic "step edges" [145].

We used 2D circularly symmetric Gaussian functions and obtained the 1st and 2nd moment statistics of the response of a filtered patch with the steerable filter. We used 5 orientations and up to 5th order Gaussian derivative. One of the main advantages observed in our experiments is that because Steerable filters use reasonably broad Gaussians, they are somewhat insensitive to localization and orientation errors.

DAISY Descriptor

The goal of DAISY descriptor is to modify descriptors such as SIFT for more efficient computation [146]. To this end, the weighted sums of gradient norms are replaced by convolutions of the gradients in specific directions with several Gaussian filters, This provides similar invariance properties with respect to SIFT, but is much faster for point matching tasks. The DAISY descriptor can be estimated as follows [146]:

1. For an image, I, the N number of orientation maps, $\mathbf{I_i}$ are estimated, one for each quantized direction, where $\mathbf{I}_o(u, v)$ is the image gradient norm at location (u, v) for direction o if it is larger than zero, else it is equal to zero. Formally $\mathbf{I_i} = \frac{\partial \mathbf{I}}{\partial o}$, $(.)^+$ was defined as an operator such that $(a)^+ = max(a, 0)$ [146].

2. Each orientation map, $\mathbf{I_i}$, is convolved multiple times with Gaussian filters of different σ values to obtain convolved orientation maps as $\mathbf{I_o^\sigma} = \mathbf{g}_\sigma * (\partial \mathbf{I}/\partial \mathbf{o})^+$ with g_σ being the gaussian filter. σ is a parameter that controls the size of the region or neighborhood. As shown in Figure 2.18, at each pixel location, DAISY's feature vector consists of a set of entries resulting from the convolved orientation maps located on concentric circles centered on the point of interest. As in the case of SIFT, a Gaussian smoothing operation is performed to decrease the effect of noisy samples located far away from the point of interest.

3. $\mathbf{h}_\sigma(u,v)$ represent the values at location (u,v) in the orientation maps after convolution by a Gaussian kernel of standard deviation σ:

$$h_\sigma(u,v) = [I_1^\sigma(u,v), \cdots, I_N^\sigma(u,v)]^T \tag{2.48}$$

4. The final description is formed by [146]:

$$\begin{aligned} f_{DAISY} = [&\check{\mathbf{h}}_{\sigma_1}^T(u_0,v_0), \\ &\check{\mathbf{h}}_{\sigma_1}^T(\mathbf{l}_1(u_0,v_0,d_1)), \cdots, \check{\mathbf{h}}_{\sigma_1}^T(\mathbf{l}_T(u_0,v_0,d_1)) \\ &\check{\mathbf{h}}_{\sigma_2}^T(\mathbf{l}_1(u_0,v_0,d_2)), \cdots, \check{\mathbf{h}}_{\sigma_2}^T(\mathbf{l}_T(u_0,v_0,d_2)) \\ &\cdots \\ &\check{\mathbf{h}}_{\sigma_Q}^T(\mathbf{l}_1(u_0,v_0,d_Q)), \cdots, \check{\mathbf{h}}_{\sigma_Q}^T(\mathbf{l}_T(u_0,v_0,d_Q))]^T \end{aligned} \tag{2.49}$$

where $\check{\mathbf{h}}_\sigma(.,.)$ represents the normalized version of $h_\sigma(u,v)$, and $\mathbf{j}_1(u_0,v_0,d_Q))$ is the location with distance d from (u,v) in the quantized direction given j [146].

Gabor Local Features

Similarly to the Tamura local features, we used 1st and 2nd statistic moments from energy of local neighborhoods around the points of interest of the image filtered with bank of Gabor filters as in Equation 2.29.

Local Color

As mentioned earlier, in object classification tasks the distinctiveness of the selected local descriptors is of critical importance. It defines the precision of the representation and the discrimination of the local features. Little work has been done in incorporating color into local features. It is important to exploit color information when describing local features. The goal is to characterize the detected points by incorporating statistical measures of salient color information around a neighborhood

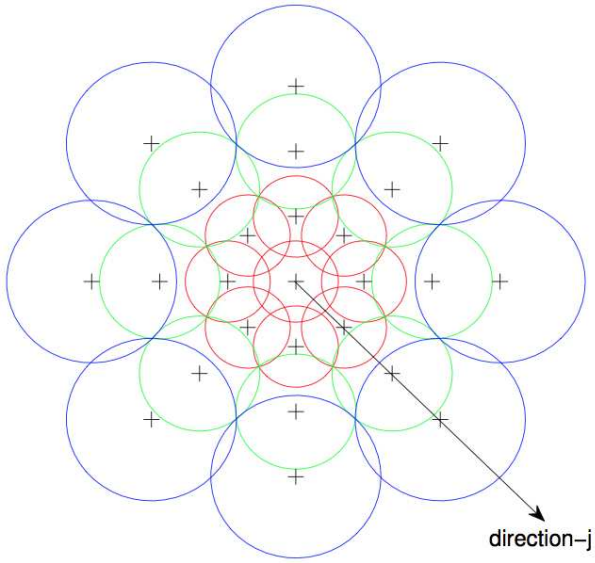


Fig. 2.18. The DAISY Descriptor. Each Circle Represents A Region Where The Radius Is Proportional To The Standard Deviations Of The Gaussian Kernels. The + Sign Represents The Locations Where The Convolved Orientation Maps Center Are Sampled. Around These Locations The Descriptor Is Estimated [146].

of points. In addition to the SIFT descriptors in the RGB color space, we examined another type of local color statistics namely *Local Color Statistics*.

Local Color Statistics: consists of estimating 1st and 2nd order statistic moments of the $R, G, B, Cb, Cr, a^*, b^*, H, S, V$ color space components of the neighboring pixels around the point of interest. The neighborhood was a square of size 16×16 oriented along the dominant orientation. Note that this feature is the local representation of the *global color statistics* described in Section 2.2.1.

The Sum of The MPEG-7 Edge Descriptor

For this descriptor the first step is to create a neighborhood around the point of interest. We selected a square neighborhood with size equal to 16×16 pixels. A neighborhood of small size is preferable for both computational cost and accuracy. The square region is oriented along the dominant orientation around the point of

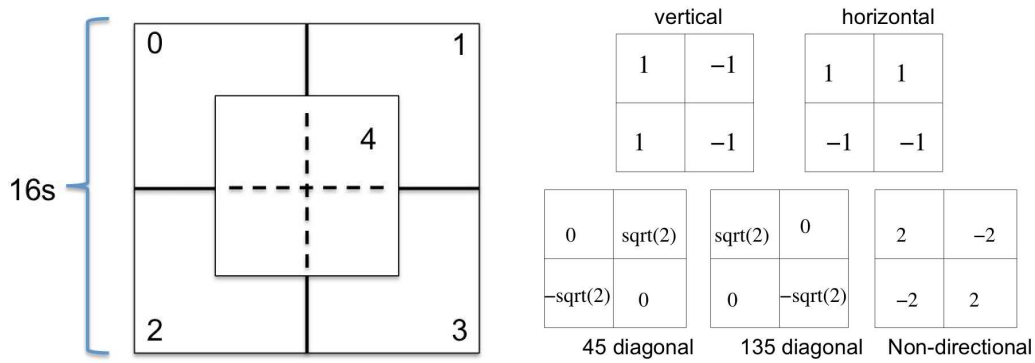
interest. Figure 2.6 shows details on how the dominant orientation is estimated for a given pixel's neighborhood. The neighborhood is subdivided in 5 square regions (top-left, top-right, bottom-left, bottom-right, and center), each of size 8×8 pixels as illustrated in Figure 2.19.a. Using the strategy presented in [123], at each region the filter response is estimated for each of the edge filter kernels proposed in the MPEG-7 standard [87], Figure 2.19.b. The sum of the response in all 5 directions are used. In order to include information about the polarity of the intensity changes, the sum of the absolute values of the responses are also estimated. The responses are normalized to the unit in order to achieve invariance to contrast. The final feature vector for each point of interest is expressed as:

$$f_m^{(i)} = (\sum d_x^{(i)}, \sum |d_x^{(i)}|, \sum d_{45}^{(i)}, \sum |d_{45}^{(i)}|, \sum d_y^{(i)}, \sum |d_y^{(i)}|, \\ \sum d_{135}^{(i)}, \sum |d_{135}^{(i)}|, \sum d_{non}^{(i)}, \sum |d_{non}^{(i)}|)_m \tag{2.50}$$

where $f_m^{(i)}$ is the feature vector corresponding to the ith region of the neighborhood of the mth point of interest, $d_x^{(i)}, d_{45}^{(i)}, d_y^{(i)}, d_{135}^{(i)}, d_{non}^{(i)}$ are the responses to the horizontal, $45\,^\circ$, vertical, $135\,^\circ$ degree, and non-directional filters. To increase the robustness with respect to geometric deformations and localization errors, the responses are first weighted with a Gaussian centered at the interest point. Therefore, the final feature vector $\mathbf{f^{(m)}}$ for a point of interest can be formulated as:

$$\mathbf{f^{(m)}} = [f_m^{(0)}, f_m^{(1)}, f_m^{(2)}, f_m^{(3)}, f_m^{(4)}] \tag{2.51}$$

This method shares a common property with the SURF descriptor, which is different from the SIFT and DAISY descriptors. That is that this descriptor integrates the gradient information within a sub neighborhood, whereas SIFT and DAISY depends on the orientations of the individual gradients. This makes this descriptor less sensitive to noise. The best performance of this descriptor was obtain when using d_x and d_y as in the case of SURF [123].



(a) Neighborhood Division Into 5 Regions (b) Edge Filters

Fig. 2.19. Filter Definitions. (a) Neighborhood Division. (b) Filters Coefficients Used.

Local Gradient Orientation Spatial Dependence Matrix (local-GOSDM)

The original GOSDM was used as a global feature, in Section 2.2.2 we showed how its design makes it invariant to rotation, but 'only' robust to scale variations by incorporating information at different scales. By turning GOSDM into a local feature, we can provide scale invariance, since the GOSDM was estimated in a neighborhood of size proportional to the most stable scale s, at which the point was detected. As in the case of the global GOSDM presented in Equation 2.2.2, we use information from the orientation gradients, in this case, around a point of interest, instead of over the entire segmented region as in the global case. As before, we are interested in describing the spatial dependence of the gradient orientations in terms of co-occurrence probability given an interest point, p_m, and the scale s. We formed a quantized version of the gradient orientation with 12 levels with respect to the dominant orientation of the point of interest neighborhood. Finally, we estimated each GOSDM using an offset equal to $1s$, $2s$, and $4s$. Four angular directions $(0\,^\circ, 45\,^\circ, 90\,^\circ, 135\,^\circ)$ were used for each offset magnitude. As a result of this process, 12, 12×12, GOSDMs were estimated. Finally, following the approach for the global GOSDM we used *Correlation, Angular Second Moment, Entropy, Contrast, and Homogeneity* statistics from each matrix.

Table 2.1
List of feature channels investigated and their type (global color, global texture or local).

Feature Channel	**Feature Type**	**Dimension**
Global Color Statistics	Global Color	20
Entropy Color Statistics	Global Color	6
Predominant Color Statistics	Global Color	28
GOSDM	Global Texture	60
EFD	Global Texture	120
GFD	Global Texture	120
SIFT	Local	128
Red-SIFT	Local	128
Green-SIFT	Local	128
Blue-SIFT	Local	128
SURF	Local	128
Tamura Descriptors	Local	3
DAISY	Local	200
Gabor Descriptor	Local	48
Local Color Statistics	Local	20
Sum of MPEG-7 Edge Descriptor	Local	128
Local GOSDM	Local	60

Table 2.1 summarizes the features described in this section, and whether they are local or global features.

3. OBJECT CLASSIFICATION

3.1 Background

Machine learning is devoted to the formal study of learning systems. There are four different types of machine learning approaches [147]:

- *Supervised learning*: in supervised learning the system is given a sequence of desired outputs $o_1, o_2, \ldots$ (training data) and the goal is to learn to produce the correct output given a new testing data sample [148].

- *Unsupervised learning*: in unsupervised learning the system receives inputs $i_1, i_2, \ldots$, but it does not contain the supervised target outputs, i.e. no groundtruth information is available. Two very simple classic examples of unsupervised learning are clustering and dimensionality reduction [149].

- *Reinforcement learning*: in reinforcement learning the system interacts with its environment by producing actions $a_1, a_2, \ldots$. The system receives rewards (or punishments) $r_1, r_2, \ldots$ based on the actions. The goal of the system is to learn to generate actions in such a way that maximizes the future rewards it receives [150].

- *Game theory* learning: here again the system receives inputs $i_1, i_2, \ldots$, produces what is known as actions $a_1, a_2, \ldots$, and receives rewards $r_1, r_2, \ldots$ based on the actions. However, the system interacts with other systems that also receive rewards based on their generated actions. The goal of each system is to maximize its rewards taking into account the other systemss current and future actions [151].

In chapter 2, we modeled the visual appearance of a segmented object by features and formed feature vectors, f. In this chapter, we propose a system to classify objects,

from segmented input images, given training segmented regions and their feature vectors (supervised learning) using several machine learning strategies. Note that we distinguish between training segmented regions and testing segmented regions. Each class/label (food class) is composed of many training objects, S_t, and their associated feature vectors, f_t. Testing data refers to segmented regions, S_q obtained from the input image, I_q, that the system has yet to classify after forming testing feature vectors f_q at the feature extraction stage.

We are interested in a system that can classify segmented regions based on their global and local properties (features) using the minimum amount of training data. The components of our system shown in Figure 3.1 are:

- Global and local feature extraction (Chapter 2)
- Individual feature channel classification - by feature channel we mean each feature space (e.g. EFD or SIFT) is classified separately and then combined or fused later as discussed below (Section 3.2)
- Visual vocabulary construction and refinement - the local features are grouped together to form a "vocabulary" of features (Section 3.2.3)
- Late decision fusion: we combine the class decision from the individual feature channel classifiers (Section 3.3)
- Contextual refinement: we correct the decisions based on contextual information (e.g., exploiting the fact that mashed potatoes with green beans occur frequently) (Section 3.4). Context refers to any information that is not directly produced by the visual appearance of an object. In this thesis we use the likelihood of object/food combinations and misclassification rates as context.

3.2 Individual Feature Channel Classification

In this section we describe how each feature channel l, $l = 0, \cdots, L$, with L the total number of feature channels in the system (e.g. *global color statistics*, *EFD*, or

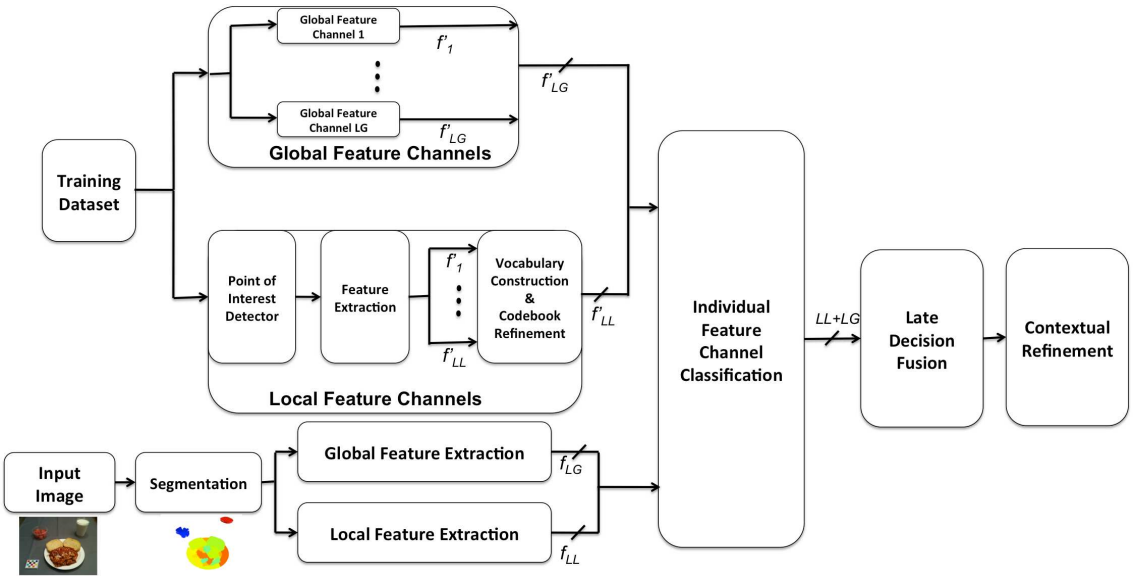


Fig. 3.1. Food Classification System. (LG is the number of global feature channels, and LL is the number of local feature channels. $f'(.)$ corresponds to the training feature set, and $f(.)$ corresponds to features of the image.)

SIFT are examples of feature channels), is classified. We refer to this as the individual feature channel classification. Given a feature vector, $f_q^{(l)}$, from a feature channel, l, extracted from input segmented regions (testing data), S_q, from the input image, I_q, a classifier assigns to the segmented region the most likely class given the model learned in the training stage using the training feature vectors, $f_t^{(l)}$, available for channel l.

In this thesis, we have studied two classifiers, *k-Nearest Neighbor* and *Support Vector Machines*, to classify the global feature channels (*global color statistics, entropy color statistics, predominant color statistics, EFD, GFD, and GOSDM*), and the *Bag-of-Features* paradigm in order to classify features extracted locally (local feature channels).

3.2.1 k-Nearest Neighbor (KNN)

The KNN classifier consists of assigning a class/label based on feature proximity (distance) in the n-dimensional feature space. KNN uses a distance measure (e.g., euclidean distance), in the feature channel, l, between the testing feature vector, $f_q^{(l)}$, and each of the training feature vectors, $f_t^{(l)}$, and then, selects the class with at least K of these training feature vectors closest to the testing feature vector [152]. KNN bases its decision locality (only the K closest training feature vectors are considered).

KNN can be seen as an online classification approach in the sense that it constructs the learning model (e.g. it estimates distances between the testing and training feature vectors) only when there is a new segmented region to be classified. There is no formal training stage. One immediate disadvantage of this approach is its large storage requirements (distances between testing feature vectors and all training feature vectors need to be obtained). Another disadvantage is that KNN is highly susceptible to what is known as the 'curse of the dimensionality'. The curse of dimensionality states that "when the dimension of a feature vector increases, the volume of the feature space increases so fast that the available data becomes sparse, and it increases the difficulty of grouping data with similar properties" [153].

The selection of K (number of closest neighbors) is critical. A large value of K contradicts the principle of locality for the class decisions since farther data points (e.g. large intraclass variation) are taken into account, and in addition, it increases the computational burden. A small value of K incorporates noisy feature vectors (e.g. feature vectors extracted from blurred images) that can lead to misclassifications. Using cross validation, we have chosen K such that it was chosen to be half of the number of total training segmented regions of the class.

3.2.2 Support Vector Machine (SVM)

The goal of a Support Vector Machine (SVM) is to produce a classification model by constructing an N-dimensional hyperplane that optimally separates the training data (feature vectors) into classes or feature space partitions [56].

SVM was first proposed for the two-class problem. This can be formulated as follows: given a training set of data pairs, (f_i, λ_i), $i = 1, ..., l$ where $f_i \in R^n$, is the n-dimensional training feature vectors and λ_i represents the class labels of the two classes (e.g. for class 1 $\lambda = -1$,and for class 2 $\lambda = 1$, or for class 1 $\lambda = 1$,and for class 2 $\lambda = 2$), SVM solves the following optimization problem:

$$\begin{gathered} min_{w,b,\zeta} \ \ (\tfrac{1}{2}\mathbf{w}^t\mathbf{w}) + C\sum_{i=1}^{l}\zeta_i \\ subject\ to \ \ \lambda_i(\mathbf{w}^t\phi(f_i) + b) \geq (1 - \zeta_i) \\ \zeta_i \geq 0, i = 1, \ldots, l \end{gathered} \tag{3.1}$$

where $\mathbf{w}$ denotes the normal vector to the hyperplane, and it is defined as $\mathbf{w} = \sum_{i=1}^{n} \alpha_i\lambda_i\mathbf{f_i}$. $C > 0$ $(C = 1/l)$ is known as penalty parameter. In this optimization problem, training feature vectors, f_t, are mapped into a higher dimensional space by the kernel ϕ so that SVM can successfully find a linear separating hyperplane in this higher dimensional space. Therefore, the solution of the above optimization problem (Equation 3.1) is given by [56]:

$$\hat{\lambda} = sgn(\sum_{i=1}^{n} \lambda_i\alpha_i K(\mathbf{f_t}, \mathbf{i}, f_q) + b) \tag{3.2}$$

where $\hat{\lambda}$ represents the predicted class that is assigned to our testing feature vector, f_q.

Obviously, this approach only works when we have two classes. However, in our case we have a multi-class problem. In this thesis, we used the "one against one" approach, where $(|\Lambda|(|\Lambda| - 1))/2$ support vector models are constructed with $|\Lambda|$ being the total number of classes in the training dataset. Each support vector model

trains data from two different classes, λ_i and λ_j, at one time [154,155] by solving the following two-class classification problem:

$$\begin{gathered} min_{w^{ij},b^{ij},\zeta^{ij}} \ \ (\tfrac{1}{2}\mathbf{w^{ij}}^t\mathbf{w^{ij}}) + C\sum_t(\zeta_t^{ij})_t \\ subject\ to\ \ (\mathbf{w^{ij}}^t\phi(\mathbf{x_t}) + b^{ij}) \geq (1-\zeta_t^{ij}),\ \text{if } \mathbf{x}_t \text{ in the i}^{\text{th}} \text{ class,} \\ subject\ to\ \ (\mathbf{w^{ij}}^t\phi(\mathbf{x_t}) + b^{ij}) \leq (-1+\zeta_t^{ij}),\ \text{if } \mathbf{x}_t \text{ in the j}^{\text{th}} \text{ class,} \\ \zeta_t^{ij} \geq 0, \end{gathered} \tag{3.3}$$

with ζ_i being variables that are related to the margin commonly introduced in this type of optimization problem [156]. In this context, the margin refers to the largest distance from the hyperplane to the nearest training feature vector of any class.

After all the two-class problems are solved, each binary classification (each two-class problem) is considered to be a "vote." The final decision is designated to be in the class with maximum number of votes.

By use of a kernel function expressed as $K(x_i, y_j) = \phi(x_i)^t\phi(x_j)$, we can reduce the search space of parameter sets. We have investigated two nonlinear kernels, namely the *Polynomial* and *Radial Basis Function* (RBF) in order to train the datasets. The formulation of the polynomial kernel is as follows:

$$K(x_i, x_j) = (1 + x_i \cdot x_j)^d \tag{3.4}$$

Where d determines the degree of the kernel. With $d = 2$ gives the quadratic kernel used in our experiments.

The RBF kernel is defined as:

$$K(x_i, x_j) = e^{-\gamma||x_i - x_j||^2} \tag{3.5}$$

In addition, we considered that C and γ are the same in all binary problems and they were set to the inverse of the number of feature vectors [56]. In our experiments we used the publicly available SVM toolbox [157].

3.2.3 Bag-of-Features (BoF)

KNN and SVM are two classifiers we used in order to classify the feature vectors of each global feature channel (*global color statistics, entropy color statistics, predominant color statistics, EFD, GFD, and GOSDM*). In this section, our approach for classifying local features is described. In Chapter 2, we described local features as the result of detecting points of interest in the image and using features around a local neighborhood around the point of interest. We have based our local feature channel classification approach on the Bag-of-Features (BoF) model due to its efficiency [85, 137, 158]. We have investigated the following local feature spaces: *SIFT, Red-SIFT, Green-SIFT, Blue-SIFT, SURF, Tamura local descriptors, DAISY, Gabor local descriptors, local color statistics*, and *local GOSDM*. Each of these feature spaces is individually classified following the BoF model.BoF computes the distribution of visual words found in the input image and compares this distribution to those found in the training images. A visual word is constructed by clustering or grouping together local feature vectors of the same feature channel. Each group or cluster of local feature vectors is referred to as a visual word. Then the distribution of visual words in the input image is estimated (e.g. how many times each visual word in the vocabulary occurs in the input image). This distribution is known as the signature of the object [51]. The final step is to use a multi-class classifier (e.g. KNN or SVM), where the signatures are the actual inputs of the classifier instead of raw feature vector values. Figure 3.3 illustrates the idea behind BoF approach. The main advantages of this approach are its simplicity, its computational efficiency, and its robustness to affine transformations, occlusions, lighting and intra-class variation [51].

Visual Vocabulary Construction

The visual vocabulary is formed by using unsupervised learning, usually clustering. Most clustering or vector quantization approaches are based on iterative square-error partitioning. These approaches attempt to obtain the feature space partition which

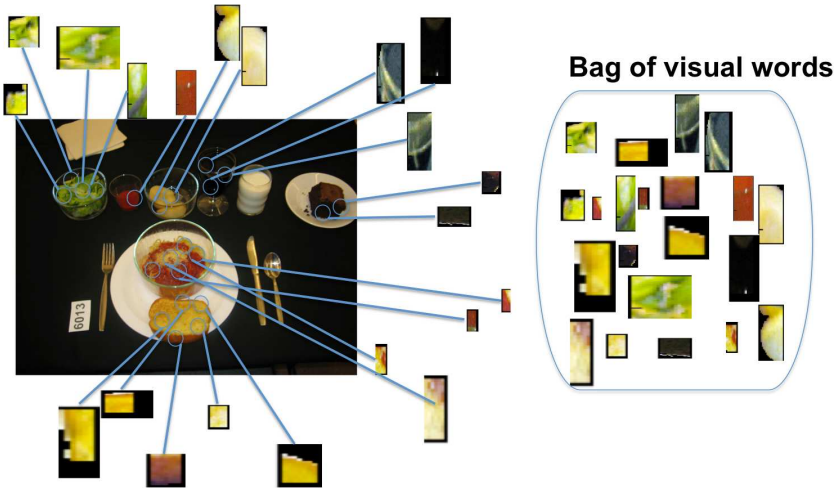


Fig. 3.2. Bag Of Visual Words Obtained From Local Features.

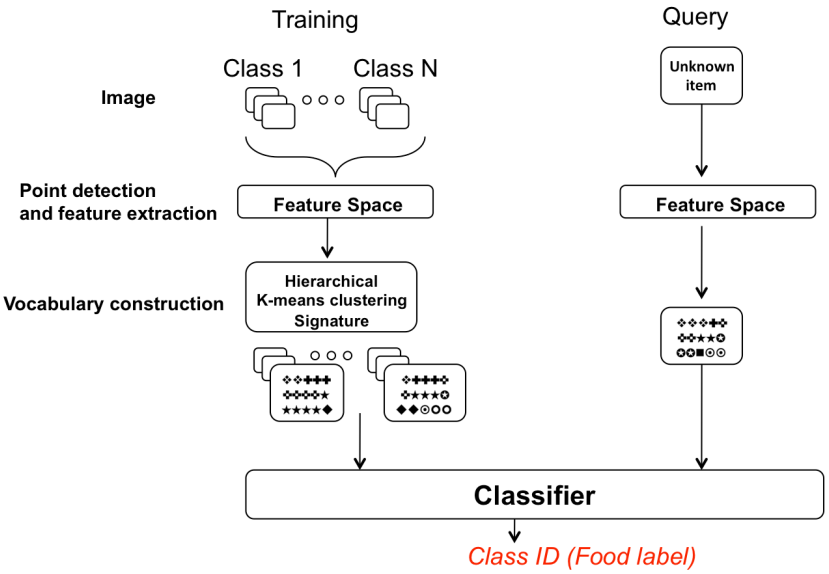


Fig. 3.3. Bag Of Features Model.

minimizes the within-cluster scatter or maximizes the between-cluster scatter of each cluster. Within-cluster scatter is a measure of how scattered samples belonging to the same cluster (or class) are with respect to each other. Between-cluster scatter measures the degree of scatter of each cluster relative to the centroids of the rest of clusters. In the BoF context, k-means is a widely used approach due to its simplicity

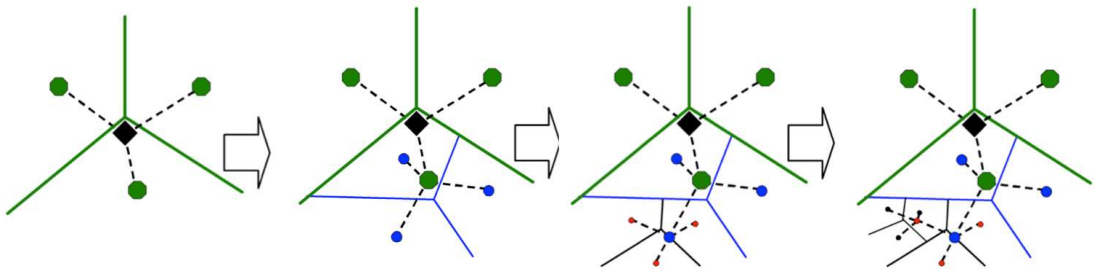


Fig. 3.4. 3 Full Hierarchical k-means Propagations [160].

[159]. K-means proceeds by iterated assignments of data samples to their closest cluster centers and re-estimating the cluster center positions at each iteration. We have considered a hierarchical version of k-means, where each local feature space is recursively divided into clusters [160]. Figure 3.4 illustrates the hierarchical k-means approach for 3 full iterations.

In general, hierarchical techniques organize data in groups that can be displayed in the form of a tree [160]. In our case, the vocabulary tree defines a hierarchical quantization that is built by hierarchical k-means clustering. Instead of k defining the final number of clusters, k defines the branch factor (number of leafs of each node) of the tree at each hierarchical level.

In our case we first, construct an initial k-means process on the training data, defining k cluster centers. Then for each cluster, data is further partitioned into k more clusters, where each cluster consists of the local feature vectors closest to the cluster center. The process is recursively applied until it reaches a target number of total levels L. In the testing stage, each local feature vector of the input segmented image is propagated down the tree at each level, up to the number of levels L, comparing the descriptor vector to the k candidate cluster centers and choosing the closest one. Figure 3.5 shows an example of the visual word signature formation using hierarchical tree for a database using SIFT as the feature space for 452 food item images.

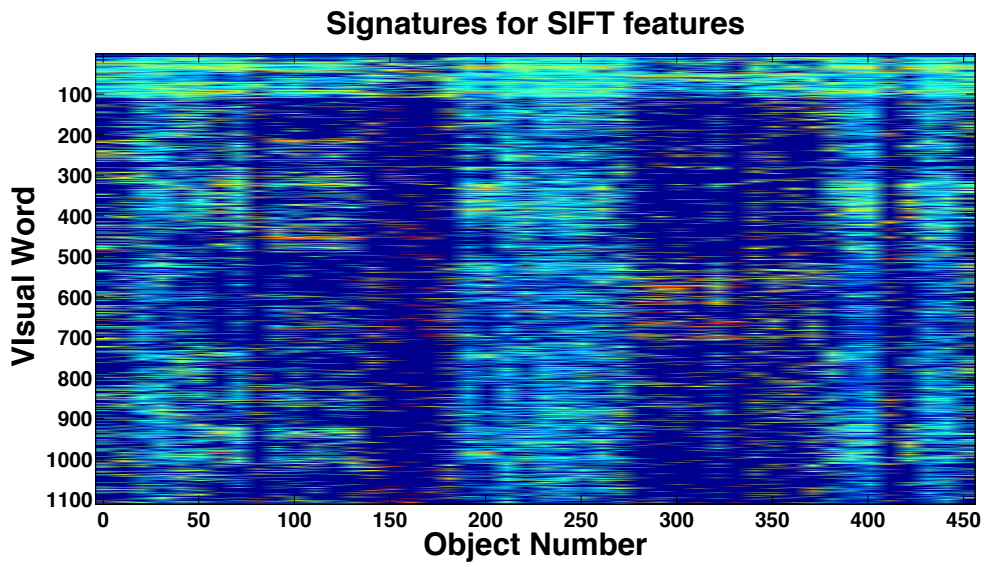


Fig. 3.5. Example Of The Visual Word Hierarchical Tree For A Database Containing 452 Objects, 1110 Visual Words, And Using SIFT As The Feature Space.

The advantages of using hierarchical k-means over one step k-means are two-fold: computational cost and new visual word handling. First, while the computational cost of increasing the size of the vocabulary in a non-hierarchical manner would be very high, the computational cost in the hierarchical approach is logarithmic in the number of leaf nodes. Second, when a new word (new local features are extracted) is added to the vocabulary, only a part of the tree will be modified, while in one-step k-means the entire vocabulary may be modified producing unstable and less robust words.

As far as the signature formation, we used two signature models. One commonly used for text categorization, *tf-idf*, and the other one based on histograms. Text categorization systems match word distributions known as *term frequencies*, n_{id}/n_d, between the input or testing and training samples, where n_{id} represents the number of times the word i occurs in the document d, and n_d is the total number of words in the document d. An *inverse document frequency weighting* $logN/N_i$ is applied to weight words that occur more frequently in the testing sample, where N_i is the

number of documents containing word i, and N is the total number of documents in the training set. In our case, the documents are the actual segmented images. *Term frequencies* and *inverse document frequency weighting* are combined to form the *term frequency-inverse document frequency* (tf-idf) measure defined as [160]:

$$\eta_i = \frac{n_{id}}{n_d} log \frac{N}{N_i} \tag{3.6}$$

Using this measure, each segmented image is represented by its signature (tf-idf vectors) as:

$$Sig_l^{(1)} = \{\eta_1, ..., \eta_i, , .., \eta_N\} \tag{3.7}$$

where $Sig_l^{(1)}$ represents the signature of the image for the $\mathrm{l^{th}}$ local feature channel, and N is the number of visual words formed. Intuitively, this signature describes the term frequency-inverse document frequency for each of the words in our vocabulary.

The second signature that has been investigated is based on histogram counts and it has the following representation:

$$Sig_l^{(2)} = \{t_1, ..., t_i, .., t_N\} \tag{3.8}$$

where $Sig_l^{(2)}$ represents the signature of the image for the $\mathrm{l^{th}}$ feature channel, t_i represents the frequency term (histogram count) of the $\mathrm{i^{th}}$ cluster. As a result of the clustering and signature formation, one signature is obtained for each testing segmented region.

The last step after the signature formation is the actual classification of the signature. There have been many strategies proposed in order to classify signatures [51, 158, 161, 162]. In our work, in order to keep consistency with global feature classification we considered KNN for its simplicity, and the kernel-based method SVM.

Codebook Refinement

A codebook refinement step can be introduced in order to guarantee the selection of representative words for each input image. Some codebook refinement strategies

include: visual word weighting (words occurring with less frequency receive high weights as a result of its uniqueness or rareness), sample strategy (detecting a large or small number of points of interest on the image), or selection of optimal vocabulary size [163].

As mentioned earlier, a visual vocabulary is generated by clustering the local descriptors or local features in their feature space and treating each resulting cluster as a visual word of the vocabulary. The number of clusters determines the size of this visual vocabulary. A small vocabulary may lack the discriminative power since two descriptors associated to different points of interest may be assigned into the same cluster even if they are not similar to each other. A large vocabulary, on the other hand, is less generalizable, and more noise sensitive. Although in general, larger vocabularies provide better results in retrieval tasks [164], it was shown that for object classification the size of the vocabulary depends on the dataset [163]. In our case, our vocabulary size ranged between 110 and 1110 visual words.

The dimensionality of the feature space may increase word ambiguity generated by noisy training data. For large dimensionality feature vectors, K-means aims at placing cluster centers near the most frequently occurring local features. As a result of this, cluster partitions are tightly found in regions with dense data distribution, and sparsely spread in sparse regions of the feature space. This can generate very nonuniform coding. In order to address the effect of high dimensional feature spaces, we examined Principal Component Analysis (PCA) to reduce the dimensionality of the features prior to form the visual vocabulary.

PCA is an orthogonal transformation used to convert a set of correlated random variables into a set of uncorrelated variables called principal components [165]. The first principal component has as high variance as possible, and each succeeding principal component has the highest variance possible while satisfying the orthogonality condition with all the preceding components. The main steps in PCA are:

1. Mean subtraction: for PCA to work properly, the mean from each of the feature vector dimensions has to be subtracted. The mean is estimated as the average value across each dimension of the local feature vector.

2. Covariance matrix estimation: the covariance matrix is defined as:

$$\Sigma = [E(X_i - \mu_i)(X_j - \mu_j)] \tag{3.9}$$

   where X_i, and X_j two random variables (in our case local feature vectors).

3. SVD Decomposition (eigen decomposition): singular value decomposition is used to obtain eigenvectors and eigenvalues of the covariance matrix Σ as: $\Sigma = UDV^*$. U and V are the orthonormal basis (eigenvectors), and D is a matrix whose main diagonal entries correspond to the eigenvalues.

4. Component selection: we select the L' eigenvectors associated to the highest eigenvalues (principal components) in order to achieve dimensionality reduction.

Trying to refine the codebook, we examined visual word formation by using spectral clustering methods instead of k-means clustering. Spectral clustering denotes a family of techniques that rely on the eigen decomposition of a modified similarity matrix to project the feature vector values prior to clustering [166,167]. In [168], a novel spectral clustering method known as Kernel Entropy Component Analysis Clustering (KECA) was proposed. KECA projects the feature vectors data by exploiting source statistics using Renyi entropy. Renyi entropy is a generalization of Shannon entropy (Equation 2.10). The Renyi entropy of order α can be defined as:

$$H_\alpha(f) = \frac{1}{1-\alpha}\log(\sum_{i=1}^{n} p_i^\alpha(f)) \tag{3.10}$$

where p_i is the probability of the feature vector values $f_1, f_2, ..., f_n$.

Once the feature vectors have been projected onto a subset of Renyi entropy, an angular distance-based cost function is used to cluster the projected data [168]:

$$J(C_1, \ldots, C_C) = \sum_{i=1}^{C} N_i \cos(\angle(\mathbf{m_i}, \mathbf{m})) \tag{3.11}$$

where C_i is the ith cluster, and $\mathbf{m_i}$ is the kernel feature space cluster mean vector corresponding to C_i. The KECA clustering is given by [168]:

1. Obtain $\mathbf{\Phi_{eca}}$ by kernel ECA as $\mathbf{\Phi_{eca}} = \mathbf{D_k^{1/2}E_k^T} : min_{\lambda_1,\mathbf{e_1},\ldots,\lambda_N,\mathbf{e_N}}[||\mathbf{m}||^2 - ||\mathbf{m_{eca}}||^2]$. Where $\mathbf{D}$ represents the diagonal matrix storing the eigenvalues $\lambda_1,\ldots,\lambda_N$, and $\mathbf{E}$ is a matrix with the corresponding eigenvectors $\mathbf{e_1},\ldots,\mathbf{e_N}$. $\mathbf{E}$ and $\mathbf{D}$ are obtained from $\mathbf{K} = \mathbf{EDE^T}$, where $\mathbf{K}$ is the kernel matrix obtained from the Renyi entropy [168].

2. Initialize means $\mathbf{m_i}$, $i = 1,\ldots,C$;

3. For all t, $\mathbf{x_t}$ is estimated as: $\mathbf{x_t} \rightarrow C_i : max_i cos(\angle(\phi_{eca}(\mathbf{x_t},\mathbf{m_i})))$

4. Mean vectors are updated

5. Steps 3 and 4 are repeated until convergence

3.3 Combination Of Local And Global Features

An important step in the design of object classification systems is the identification of relevant class specific features. In Chapter 2, we described many feature spaces that follow this idea. Besides the design of different discriminative feature channels, their combination in order to obtain a final decision needs to be addressed.

Generally speaking, the output information that a classifier supplies can be divided into three levels [169]:

1. The abstract level: A classifier, ϑ_l, only outputs a class label λ_l, or a subset $\lambda^* \subset \Lambda$.

2. The rank level: The classifier, ϑ_l, ranks all the labels in Λ, or the subset $\lambda^* \subset \Lambda$, with the label at the top being the first choice.

3. The measurement level: The classifier, ϑ_l, generates a measurement value to address the degree that an object/segmented region, S_q, has the label λ for each label in Λ or for a subset $\lambda^* \subset \Lambda$.

Based on these definitions we combined and fused the outputs of the individual feature channel classifiers for a final class decision. Hence, we are individually classifying each feature space, and combining classifier decisions (the abstract level) along with confidence scores (the measurement level) given the top β candidate classes (the rank level) for each feature channel. By candidate classes, we mean the top β class labels that a classifier selects and ranks given an input segmented image I_q.

3.3.1 Classifier Confidence Measure

As previously mentioned not only each feature channel classifier decision is considered, but also the confidence score from such classifier. The confidence score describes the classifier's confidence that its inferred label is correct. When the KNN classifier is used, the confidence score $\phi(S_q, \lambda)$, for assigning segmented region S_q of an input image I_q to class λ in the feature channel l is defined as:

$$\phi_l(S_q, \lambda)^{(k-NN)} = \frac{1}{k}\sum_{i=1}^{k} exp(-d(f_{S_q}, f_{S^i_\lambda})/(d_{1-NN} + \epsilon)), \text{for each } \lambda \in \Lambda \tag{3.12}$$

where $d(f_{S_q}, f_{S^i_\lambda})$ represents the distance between normalized feature vector of the input segmented region S_q, f_{S_q}, and the normalized feature vector of the ith nearest neighbor belonging to class λ, $f_{S^i_\lambda}$. d_{1-NN} represents the distance between normalized feature vector of the input segmented region, S_q, and the nearest neighbor (1-NN). We added ϵ to denominator to avoid the case of division by zero. Λ is the collection of class labels in the dataset.

When SVM is used, then the classifier's confidence score is:

$$\phi_l(S_q, \lambda)^{(SVM)} = exp(-d(f_{S_q}, f^{(ave)}_{S^i_\lambda})/(d_{1-NN} + \epsilon)), \text{for each } \lambda \in \Lambda \tag{3.13}$$

where $d(f_{S_q}, f^{(ave)}_{S^i_\lambda})$ represents the distance between normalized feature vector of the input segmented region S_q, f_{S_q}, and the normalized average feature vector for class λ, $f^{(ave)}_{S^i_\lambda}$. For the local features, the average class feature vector, $f^{(ave)}_{S^i_\lambda}$, is a feature vector containing the frequency that each visual word is observed in class λ on average.

3.3.2 Late Decision Fusion

As a result of individual feature channel classification, for each segmented region, S_q, and feature channel, l, a set of β candidate class labels, $\lambda_l^{cand} = [\lambda_l^1(S_q), \cdots, \lambda_l^\beta(S_q)]$, and confidence scores $\phi_l^{cand}(S_q, \lambda_l^{cand}) = [\phi_l(S_q, \lambda_l^1), \cdots, \phi_l(S_q, \lambda_l^\beta)]$ are available at the output of each classifier (Figure 3.1). Our goal is to determine the final class label, $\hat{\lambda}(S_q)$, for the segmented region, S_q, given the individual decisions and scores of each classifier. Two strategies are considered for the final decision:

- Maximum confidence score: this consists of obtaining the class label such that the confidence score from all the feature channel classifiers is the largest. That is, for each segmented region S_q select the class label such that it satisfies $\hat{\lambda}(S_q) = argmax_{\lambda^\star \in |\Lambda|}(\sum_{l=1}^{L} \sum_{b=1}^{\beta} \phi_l^b(S_q, \lambda_l^b)$, with L being the number of feature channels.

- Majority vote rule: we take a majority vote on the set $\lambda_1^{cand}, \cdots, \lambda_L^{cand}$. In case of same number of votes, the tie-breaker is the output of the individual classifier that achieves higher classification rate over the validation set. Majority rule can be seen as a variation of maximum confidence score for the case that all k nearest neighbors are at equal distance in the feature space from the input segmented region.

As mentioned in the introduction, these confidence scores were also used in our multiple hypothesis segmentation and classification system (MHSC), in order to obtain "optimal" segmentations of the input image [64].

3.4 Context-Based Refinement

In this section, we describe the model we investigated for using contextual information in order to reach a labeling agreement for all the segmented regions of the input image. The goal is to detect potential misclassifications and further refine the

classifier decisions obtained after late decision fusion. By contextual information we mean any information that is not directly produced by the appearance of an object.

Traditional approaches to object classification use appearance (visual) features as main source of information to assign an object to a class. Visual features face many image perturbations such as clutter, noise, variation in pose, and lighting conditions. Biederman et al. [170] proposed five different classes of object configurations that can be used as 'side' information in any classification system. These are *interposition, support, probability, position,* and *familiar size.* These characterize the organization of objects in real-world scenes. The first two, *interposition* and *support* refer to physical space. *Probability, position* and *size* are defined as semantic relations because they require access to the semantic meaning of the object. Semantic relations include information about detailed interactions among objects in the scene and they are used as contextual features.

In [61], contextual features are grouped in three categories: semantic context (probability), spatial context (position), and scale context (size). Semantic context was modeled by a binary co-occurrence matrix that relates objects i and j in a scene [171]. They used Google Sets to derive semantic context.

Fischler in [54] proposed a scheme to recognize various objects and the scene. Classification was done by segmenting the image into regions, labeling each segmented region as an object and refining object labels using spatial context as relative locations. Refining objects was described by breaking down the object into a number of more "primitive parts" and by specifying an allowable range of spatial relations which these "primitive parts" must satisfy for the object to be present. Other approaches incorporated spatial context from inter-pixel statistics and pairwise relations between regions in images.

Spatial and scale context are the most exploited types of context by classification systems. Semantic context, however, it is the only context type that brings out the most valuable information for improving classification. Considering the variability of the object and food item configurations in the scene, scale and spatial relations vary

in greater extent than the co-ocurrence of objects (e.g. there is no guarantee that two food items will always have the same spatial position with respect to one another). Co-occurrences are much easier to access than spatial or scale relationships and much faster to process and estimate.

A common model to represent contextual information are Conditional Random Fields [172, 173].

Conditional Random Fields: Conditional Random Fields (CRF) are motivated by Hidden Markov Models (HMM). A HMM joint distribution $p(X, Y)$ can be written as [173]:

$$p(X,Y) = \frac{1}{Z}\exp\{\sum_r \sum_{i,j\in S} \xi_{ij}\mathbf{1}_{y_r=i}\mathbf{1}_{y_{r-1}=j} + \sum_r \sum_{i\in S}\sum_{o\in O} \xi_{oi}\mathbf{1}_{y_r=i}\mathbf{1}_{x_r=o}\} \tag{3.14}$$

where ξ_{ij}, ξ_{oi} model the distribution, $p(X, Y)$. $\xi_{ij} = \log p(y' = i|y = j)$ it is commonly used. Z is a constant value that ensures the probability function to sum to 1. In general a conditional distribution can be written as:

$$P(A|B) = \frac{P(A,B)}{P(B)} \tag{3.15}$$

with A and B being two random variables. From this we can write the conditional distribution $p(Y, X)$ that results from HMM (Equation 3.14). This is [173]:

$$p(Y|X) = \frac{p(Y,X)}{\sum_{Y'} p(Y',X)} = \frac{\exp\{\sum_{k=1}^{K} \xi_k f_k(y_r, y_{r-1}, x_r)\}}{\sum_{Y'} \exp\{\sum_{k=1}^{K} \xi_k f_k(y'_r, y'_{r-1}, x_r)\}} \tag{3.16}$$

where from Equation 3.14:

$$\exp\{\sum_{k=1}^{K} \xi_k f_k(y_r, y_{r-1}, x_r)\} = \exp\{\sum_r \sum_{i,j\in S} \xi_{ij}\mathbf{1}_{y_r=i}\mathbf{1}_{y_{r-1}=j} + \sum_r \sum_{i\in S}\sum_{o\in O} \xi_{oi}\mathbf{1}_{y_r=i}\mathbf{1}_{x_r=o}\} \tag{3.17}$$

This conditional distribution (Equation 3.16) is a CRF. We can rewrite Equation 3.16 as [173]:

$$p(Y|X) = \frac{1}{Z(X)}\exp\{\sum_{k=1}^{K} \xi_k f_k(y_r, y_{r-1}, X_r)\} \tag{3.18}$$

where Z(X) is defined as:

$$Z(X) = \sum_{y} \exp\{\sum_{k=1}^{K} \xi_k f_k(y_r, y_{r-1}, X_r)\} \tag{3.19}$$

An important point is how to estimate the parameters $\Xi = \{\xi_k\}$ of a CRF. Penalized maximum likelihood is used for parameter estimation:

$$L(\theta) = \sum_{i=1}^{N} log(p(Y^i|X^i)) = \sum_{i=1}^{N}\sum_{r=1}^{T}\sum_{k=1}^{K} \xi_k f_k(y_r, y_{r-1}, X_r) - \sum_{i=1}^{N} \log(Z(X^i)) \tag{3.20}$$

In general, numerical optimization is used in order to maximize the function $L(\theta)$. The partial derivatives of (Equation 3.20) are [173]:

$$\frac{\partial L}{\partial \xi_k} = \sum_{i=1}^{N}\sum_{r=1}^{T} f_k(y_r^i, y_{r-1}^i, X_r^i) - \sum_{i=1}^{N}\sum_{r=1}^{T}\sum_{y,y'} f_k(y_r^i, y_{r-1}^i, X_r^i) p(y, y'|X^i) \tag{3.21}$$

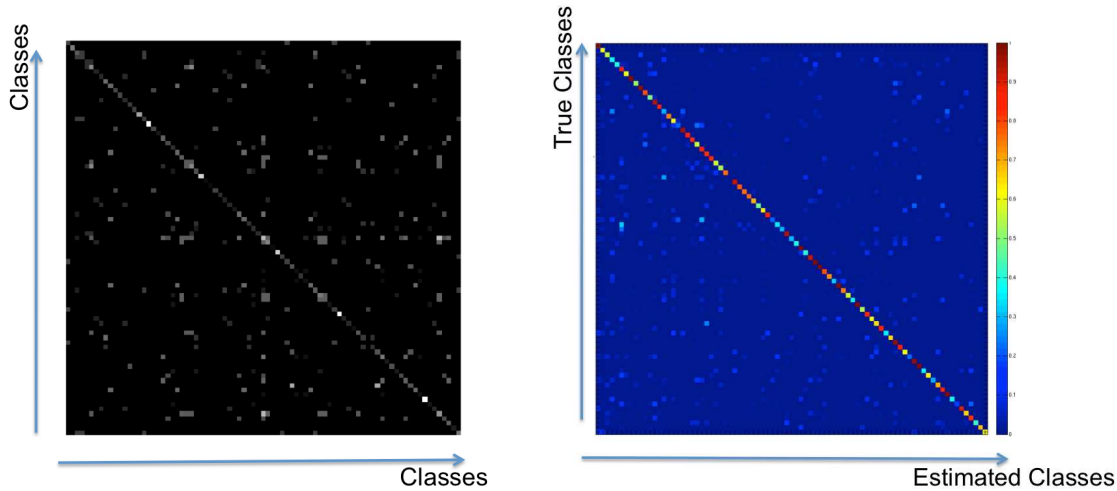
Misclassifications and Object Interaction-based Contextual Information Refinement: In this thesis, two sources of information have been considered for contextual information refinement: object interactions and misclassifications. Object interactions are modeled by the probability of occurrence of object pairs in the scene/eating occasion. Misclassifications are modeled by the confusion matrix from the classifiers. Figure 3.6 illustrates an example of object likelihood combinations, and misclassification for a food image database.

CRFs are used in order to take advantage of 1st and 2nd order class object interactions in the image. The CRF model maximizes the conditional distribution of the class labeling given the image.

In our problem, CRF provides a discriminative framework for modeling the probability of a particular label sequence Λ given the observation sequence χ (image), $P(\Lambda|\chi)$, as the normalized product of potential functions (1st and 2nd order potential functions) of the form:

$$P(\Lambda|\chi) = \frac{1}{Z(\chi, \xi)} \exp(\sum_{S_i \in S} A_i(\lambda_i, \chi, \xi_a) + \sum_{S_i \in S}\sum_{S_j \in S} I_{ij}(\lambda_i, \lambda_j, \xi_i, \chi)) \tag{3.22}$$

where $Z(\chi, \xi)$ is the partition function. $A_i(\lambda_i, \chi, \xi_a)$ is the association potential, or state features function, which describes how likely a segmented region S_i is to belong



(a) Object combinations (b) Confusion Matrix

Fig. 3.6. Contextual Information Examples: a) Object Combinations, b) Object Misclassifications (right).

to a label λ_i given the observed data χ while ignoring other segmented regions in the image. ξ_a is a vector to describe the weights of the 1st order potentials. $I_{ij}(\lambda_i, \lambda_j, \xi_i, \chi)$ is known as interaction potential. It measures the compatibility and consistency of the class label assignments given the observations χ. ξ_i describes the weights of the 2nd order potentials. ξ can be defined as $\xi = [\xi_a, \xi_i]$. We model $A_i(\lambda_i, \chi, \xi_a)$ as:

$$A_i(\lambda_i, \chi, \xi_a) = h(\xi_a, \Gamma(\lambda_i, \lambda_m), \Phi(i, \lambda)) = \sum_{r=1}^{L|\Lambda|} \xi_{a_r} \gamma_r(\lambda_i, \lambda_m) \phi_r(i, \lambda_i) \tag{3.23}$$

where $\Gamma(\lambda_i, \lambda_m) = [\gamma_1(\lambda_i, \lambda_1)\, \gamma_2(\lambda_i, \lambda_1) \cdots \gamma_L(\lambda_i, \lambda_1) \cdots \gamma_l(\lambda_i, \lambda_m) \cdots \gamma_L(\lambda_i, \lambda_{|\Lambda|})]$ describes the misclassification rate observed in the experiments for class λ_m with class λ_i for each feature channel l, *i.e.* the elements from the confusion matrix. $\Phi(i, \lambda) = [\phi^1_{(i,\lambda_1)} \cdots \phi^L_{(i,\lambda_1)} \cdots \phi^l_{(i,\lambda_i)} \cdots \phi^l_{(i,\lambda_m)} \cdots \phi^L_{(i,\lambda_{|\Lambda|})}]$ is the confidence measure of each feature channel classifier for class λ of segmented region S_i, $|\Lambda|$ is the total number of classes. $A_i(\lambda_i, \chi, \xi_a)$ can be seen as the potential that models how confident is a classifier that the segmented region S_i belongs to class λ.

The interaction potential describes the object interaction between two segmented regions in the testing image through the co-occurrence information as follows:

$$I_{ij}(\lambda_i, \lambda_j, \xi_i, \chi) = \lambda_i \lambda_j \xi_{ij} \theta(i, j) \tag{3.24}$$

with $\theta(i, j)$ being the probability of co-occurrence between objects of class λ_i and class λ_j in the same image learned from our labeled training data.

The final expression of the conditional probability of the most likely sequence of class labels given the image observations can be obtained by rewriting Equation 3.22 using Equation 3.23 and Equation 3.24 as:

$$P(\Lambda|\chi) = \frac{1}{Z(\chi, \xi)} \exp(\sum_{i \in S} \sum_{r=1}^{LN_m^{(\lambda_i)}} \xi_{a_r} \gamma_r(\lambda_i, \lambda_m) \phi_r(i, \lambda_i) + \sum_{i \in S} \sum_{j \in S} \lambda_i \lambda_j \xi_{ij} \theta(i, j)) \tag{3.25}$$

and,

$$Z(\chi, \xi) = \sum_{\lambda \in \Lambda} \exp(\sum_{i \in S} \sum_{r=1}^{LN_m^{(\lambda_i)}} \xi_{a_r} \gamma_r(\lambda_i, \lambda_m) \phi_r(i, \lambda_i) + \sum_{i \in S} \sum_{j \in S} \lambda_i \lambda_j \xi_{ij} \theta(i, j)) \tag{3.26}$$

Parameter learning, inference and classification refinement: Given a set of N_t i.i.d. label training images, the parameters ξ are estimated by maximizing the log-likelyhood, $\Omega(\xi) = \sum_{t=1}^{N_t} logP(\Lambda|\chi, \xi)$, where $P(\Lambda|\chi, \xi)$ is interpreted as $P(\Lambda|\chi)$ (Equation 3.25). This optimization can be intractable due to the nature of $Z(\chi, \xi)$. Therefore, the maximization of the pseudo-likelihood of the training data (sum of local likelihoods) is done. This optimization is based on an unconstrained gradient descend method, L-BFGS [174]. The final inference provides with the most likely configuration of labels λ_i (Viterbi labeling) using Maximum Posterior Marginal (MPM) estimation [172]. Expressed as $\lambda_i = \lambda(S_q) = argmax_{\lambda^\star \in |\Lambda|} P(\lambda|\chi)$. The implementation of our CRF uses the publicly available CRF toolbox [175].

4. EXPERIMENTAL RESULTS

In this chapter, we present experimental results for the classification system and the features we described earlier. Section 4.1 describes a set of experiments to examine the efficiency of the global color features described in Section 2.2.1. In Section 4.2 we present evaluations of the proposed texture features (Section 2.2.2). In Section 4.3, an evaluation for the local descriptors is presented, including point detector accuracy, and the feature efficiency.

Overall system classification accuracy is described in Section 4.4 with emphasis on the contribution of the individual feature channels, the late decision fusion efficiency, and the use of contextual information. Finally, in Section 4.5, we present results of user studies.

4.1 Color Features Evaluation

The goal of this section is to measure the efficiency of the color features proposed in Section 2.2.1.

4.1.1 Datasets Used For Evaluation Of The Color Features

For the evaluation of color description, we need to isolate the error caused by the color description from unstable automatic segmentations. For this reason, we used a collection of food categories where training and testing segmented regions have been manually segmented (groundtruth data). Manually segmented regions provided stable segmentations. The classification was performed using a nearest neighbor classifier, which is a special case of KNN for $k = 1$.

For these experiments a total of 37 food classes were selected: *Apple Juice, Bagel, BBQ Chicken, Broccoli, Brownie, Canned Pears, Catalina Dressing, Chocolate Cake, Coke, Cream Cheese, Egg (Scrambled), French Dressing, French Fries, Fruit Cocktail, Garlic Bread, Gravy Chicken, Green Beans, Hamburger, Ketchup, Lettuce, Mac and Cheese, Margarine, Mashed Potatoes, Milk, Orange Juice, Peach, Peanut Butter, Pineapple, Pork Chop, Regular Coffee, Sausage, Spaghetti, Strawberry Jam, Sugar Cookie, Toast, Vegetable Soup, Yellow Cake.* Figure 4.1 shows an example of each manually segmented food category.

4.1.2 Color Feature Performance

The correct classification percentage is used to measure the impact of the color features in object classification tasks. As mentioned earlier, for this set of experiments, each feature vector is classified based on a nearest neighbor using l_2-norm as the distance metric.

In Section 2.2.1 we described the three global color feature channels used, namely *global color statistics, entropy color statistics,* and *predominant color statistics. Global color statistics* consists of extracting the 1st and 2nd statistic moments of the following 10 color space components $R, G, B, Cb, Cr, a^*, b^*, H, S, V$, as a result a 20 dimensional feature vector is obtained. Table 4.1, first row, show the classification performance of the *global color statistics* as a function of the percentage of training data used ($25\%, 50\%$, and 75%).

To evaluate our *Entropy color statistics* feature, we repeated the experiments used to evaluate the previous color descriptor, *i.e.* a classification task using 37 manually segmented food categories. The *Entropy color statistics* consists of estimating the 1st and 2nd moment statistics of the entropy of the *R, G, B* color space components. As a result, a six dimensional feature vector is formed. The second row of Table 4.1 shows the results using this color feature. Overall *global entropy statistics* performed the worst among the three color feature channels (*global color statistics*, *global entropy*

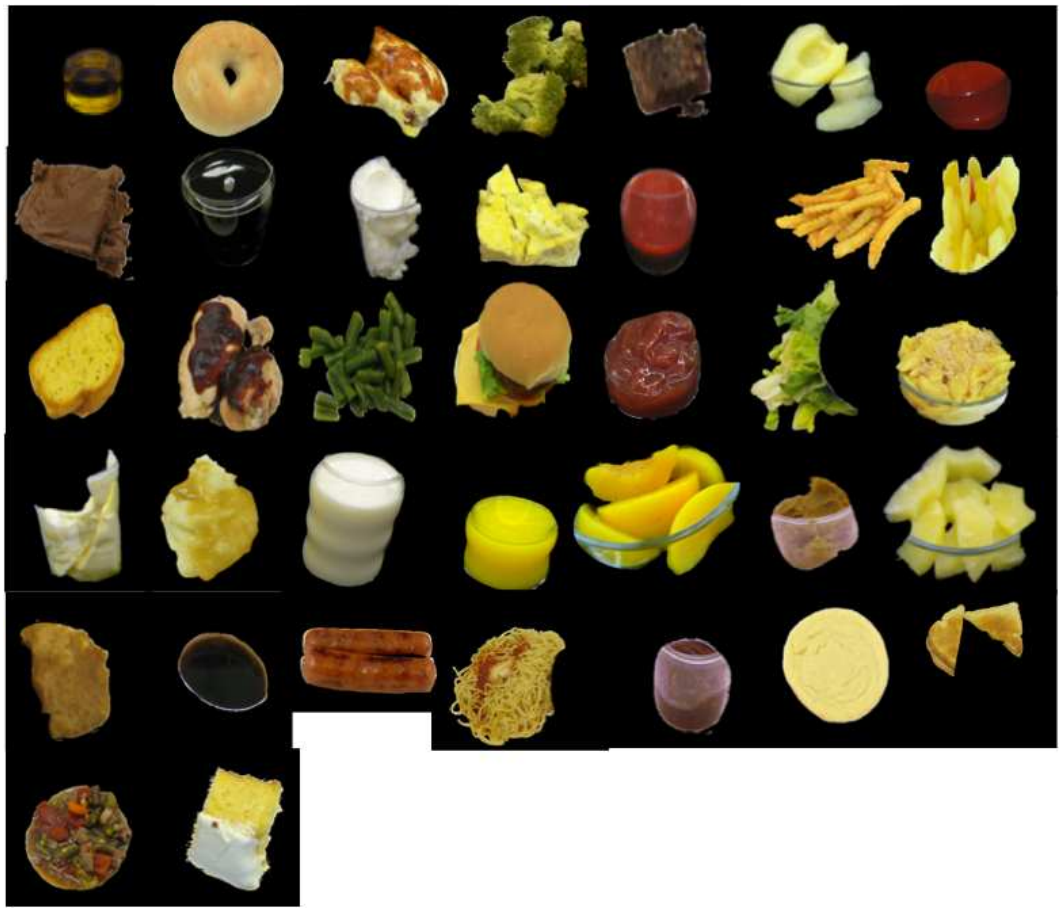


Fig. 4.1. Examples Of 37 Food Classes Used In The Color Experiments.(From left to right and top to bottom: *Apple Juice, Bagel, BBQ Chicken, Broccoli, Brownie, Canned Pear, Catalina Dressing, Chocolate Cake, Coke, Cream Cheese, Egg, French Dressing, French Fries, Fruit Cocktail, Garlic Bread, Gravy Chicken, Green Beans, Hamburger, Ketchup, Lettuce, Mac and Cheese, Margarine, Mashed Potatoes, Milk, Orange Juice, Peach, Peanut Butter, Pineapple, Pork Chop, Regular Coffee, Sausage, Spaghetti, Strawberry Jam, Sugar Cookie, Toast, Vegetable Soup, Yellow Cake*).

statistics, *predominant color statistics*), one potential reason is that this feature was designed for complex foods (e.g. *soup*) and in our database there are few complex foods. For many color homogeneous foods in our database this feature will show very similar values becoming prone to misclassifications.

Table 4.1
Average classification rate of color features using 1-Nearest Neighbor with L_2 norm. There are 37 food categories. Experiments were performed with random selection of training and testing data, as well as increasing the percentage of training data used (25%, 50% and 75%).

Color Feature	**Dimension**	**Average Classification 25% training**	**Average Classification 50% training**	**Average Classification 75% training**
1st and 2nd moments of R,G,B,a,b,Cb,Cr,H,S,V	20/object	0.64	0.78	0.83
Color entropy moments	6/object	0.36	0.46	0.55
Predominant color statistics	28/object	0.54	0.60	0.65

In order to be consistent in all our color feature channels experiments, we repeated the same experiment for our last global color feature channel, *Predominant color statistics*. It consists of estimating the P most representative colors (in RGB space) for a segmented region. The third row of Table 4.1 shows the correct classification results obtained with the predominant color. Note that for these experiments the number of predominant colors used for each food item was set to 4 ($P = 4$). Small $P = 1, 2$ would not provide enough discriminative information about the object. A large choice of the parameter P was not a good solution for very homogeneous food items and we may be also incorporating effects from shadows and reflections that change the visual properties of the object. A different choice of P value may be needed for other databases with objects with other color compositions [87].

Overall the *Global color statistics* outperformed the other two color global features. In part this is due to having stable segmentations of the objects (manually segmented objects). In food items with a great deal of color variation (e.g. *soup* or *burger*), using moment statistics from the estimated entropy of the R, G, B color space components was a good solution for their color description. Finally, the predominant color could be useful to account for intra-class variations or food color irregularities such as sausage with some roasted (darker) areas. In this case an average color (*Global color statistics*) of the entire object including the roasted area would not find a good match in the training set. However, when decomposing the item into predominant colors the most predominant colors are not, in general, due to any color perturbation, so correct classification was achieved. Each of the global color features proven to be useful for specific food items so we incorporated all three of them in our final food identification system.

4.2 Texture Analysis

One of the contributions of this thesis is the three texture features: EFD, GFD, and GOSDM (Section 2.2.2). A separate evaluation of the descriptors with texture

images is presented in this section. EFD (entropy-based categorization and fractal dimension estimation) categorizes a texture pattern based on its entropy properties. The fractal dimension is estimated for each point set according to this categorization. In our simulations the dimension of the EFD feature vector was set equal to 120. This was obtained by uniformly partitioning the entropy image into 4 levels, and for each entropy level estimating the FD using the counting box method for 30 levels ($\epsilon = 1, \cdots, 30$) [119]. In GFD (Gabor-based image decomposition and fractal dimension estimation), a set of Gabor filterbanks decompose the texture into categories based on their spatial frequency. There are 24 frequency distributions (4 scales and 6 orientations). For each decomposition the FD is estimated using the counting box method with 5 levels [119]. The total dimension of the GFD texture feature was equal to 120. Finally GOSDM (gradient orientation spatial-dependence matrix) describes the probability of occurrence between pairs of gradient orientations at different given offsets $d = 1, 4, 16$ and angular orientations $(1, 0), (\sqrt{2}/2, \sqrt{2}/2), (0, 1), (-\sqrt{2}/2, \sqrt{2}/2)$ using the following statistics: Angular second moment, homogeneity, entropy, correlation, and contrast. Thus, the dimensional of the GOSDM vector was equal to 60).

4.2.1 Datasets For Evaluation Of The Texture Features

In order to test the proposed texture features (Section 2.2.2), we considered three very different datasets. Our first dataset was a food texture dataset since our ultimate goal was to describe food textures [71]. Our second dataset was the well-known *Brodatz* texture dataset that contains many different types of textures [176]. Finally, a third dataset with many viewpoint and lighting changes namely *UIUC texture* dataset [177].

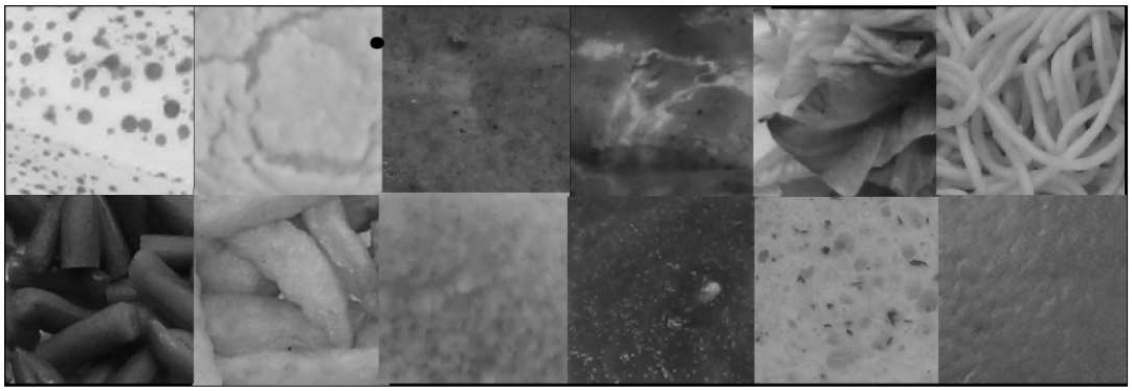


Fig. 4.2. Examples Of Our Customized Food Texture Dataset.

Food Texture Dataset

We created a dataset containing texture details of many foods. Each texture detail was selected so that texture properties were captured at different lighting conditions for each class. A total of 34 food texture classes with 10 (128×128-pixel) samples per class were used.

The Brodatz Dataset

The Brodatz texture album [176] is a well-known benchmark dataset for texture analysis. It contains 111 different texture classes where each class is represented by one (640×640 pixel) image. We divided each image into 25 (128×128 pixel) sub-images (Figure 4.3) for a total of 2775 texture images. Note that this dataset is somewhat limited, as it does not model viewpoint, scale, or illumination changes. Some of the texture classes are very inhomogeneous, which made it very difficult to model them using some of the other sub-images. Also, having 111 different classes also represented a challenge for our texture descriptors.

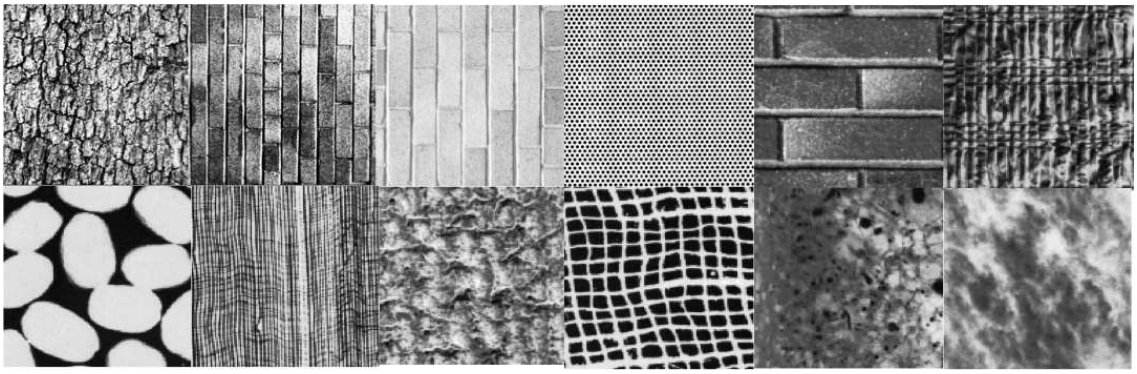


Fig. 4.3. Examples Of The Brodatz Dataset Texture Samples.

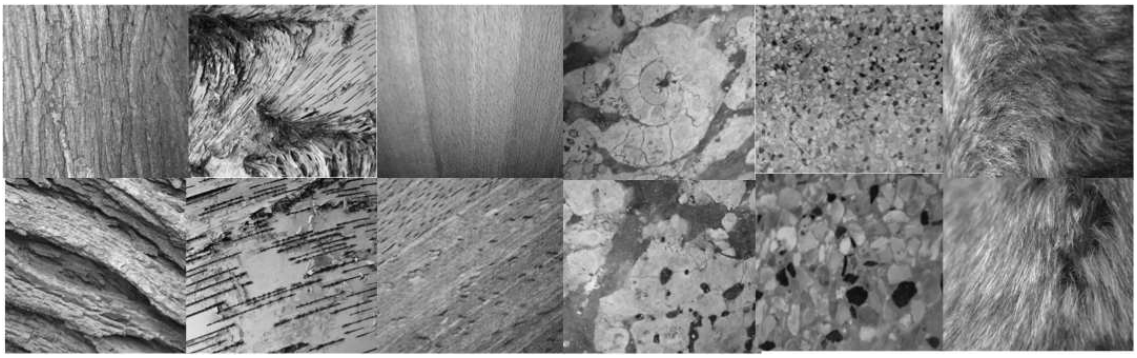


Fig. 4.4. Examples Of The UIUC Dataset Texture Samples.

The UIUC Dataset

The last dataset is very popular for testing the effectiveness of local texture features [177] since it is very challenging. It consists of 1000 images of different object texture details: 40 samples each of 25 different textures. Figure 4.4 shows two sample images (the resolution of the sample is 640×480 pixels). The database includes textures with significant viewpoint changes and scale differences within each class. The images were acquired under no illumination conditions control. Other sources of intra-class variation include non-planarity of the textured surface, significant non-rigid deformations between different samples of the same class, and even, inhomogeneities of the texture pattern.

4.2.2 Texture Descriptors Evaluation

A set of texture classification experiments were conducted to determine the efficiency of the proposed texture descriptors: EFD, GFD, and GOSDM. We were interested on comparing them with well-known texture feature approaches such as Global Gabor features [118], Gray Level Co-occurrence Matrices (GLCM) [104], and Multifractal Spectrum (MFS) features [178]. Texture descriptors were evaluated based on classification accuracy experiments.

In these experiments, we wanted to evaluate the texture descriptors in terms of classification performance. We used a distance-based classification strategy to classify each texture image. For each class of the training set, we averaged each element of the feature vector $\mathbf{f} = [f_1 \ldots f_K]$ for all the training texture samples resulting in one unique training feature vector per texture class:

$$\mathbf{F_{class_j}} = [F_1 \; F_2 \ldots F_K] = [\frac{\sum_{i=1}^{N_t^j} f_1}{N_t^j} \; \frac{\sum_{i=1}^{N_t^j} f_2}{N_t^j} \ldots \frac{\sum_{i=1}^{N_t^j} f_K}{N_t^j}]_j \tag{4.1}$$

where F_{class_j} represents the training feature vector of class j, N_t^j is the number of training images for class j, and K is the dimension of the feature vector for each texture image. The test texture image, q, is assigned to the class λ as follows:

$$\lambda = \arg \min_{\boldsymbol{j, j=1,\ldots,N_c}} d(f_q, \mathbf{F_{class_j}}) \tag{4.2}$$

where $d(.)$ is the l_2-norm, N_c number of classes, and f_q the feature vector for the test image q. Table 4.2 shows the dependence of classification rates for each of the texture descriptors on the number of training images per class for all three databases.

From the results, we can see that the fractal-based approaches EFD, GFD, outperformed the primary texture descriptor based on multifractal information (MFS) and consistently had higher classification rates. In general, our fractal-based approaches (EFD and GFD), describe very discriminative texture feature spaces, in particular for foods. Also, categorizing textures based on entropy information (EFD) instead of gray level intensity (MFS) proved to be more efficient for many types of non-food textures as well.

Table 4.2
Classification rates using GLCM, Gabor, MFS, EFD, GFD, and GOSDM for food dataset, Brodatz, and UIUC using different training data percentages (40%, and 80%). Feature vector dimension is also included.

	Food		**Brodatz**		**UIUC**	
Features (Dimension)	**40%**	**80%**	**40%**	**80%**	**40%**	**80%**
GLCM (32)	.23	.33	.41	.50	.14	.18
Gabor (48)	.51	.50	.71	.82	.19	.25
MFS (90)	.49	.51	.71	.73	.23	.35
EFD (120)	.74	.74	.88	.85	.29	.38
GFD (120)	.70	.69	.93	.91	.23	.32
GOSDM (60)	.39	.55	.68	.72	.27	.34

Comparing the Gabor-based approaches: both the GFD and the classical Gabor-like features showed good performance. However, the GFD outperformed Gabor-like features in many of the texture classes. This may indicate that 1st and 2nd statistical moments from the energy of the Gabor filtered image may not provide enough discrimination to represent textures as well as the fractal-based representations, particularly for food textures.

Finally, the GOSDM performed better than the GLCM. These are conceptually similar descriptors in the sense that the same statistics are used. The use of the gradients instead of the gray level values of the image was motivated by the robustness of the gradient against illumination changes and other distortions.

The UIUC dataset was a very challenging dataset, mainly due to the large lighting, viewpoint, and scale variations. Also the classifier selected is distance-based and is not optimal for these types of image variations. Much better results have been reported using local features based bag-of-features (BoF) instead of global texture features [177]. Global texture features showed low performance with this dataset using a distance-based classifier. However, our goal in this section was to select global textures to complement the global color features for a more complete object description. EFD, GFD, and GOSDM outperformed the other features investigated.

4.3 Local Descriptors and the Bag-of-Features (BoF) Evaluation

In this section, we describe the experiments conducted to evaluate the local descriptors and the Bag-of-Features (BoF) approach. We first compared the performance of the point detectors described in Section 2.3, namely *DoG*, and *entropy points* detector. We also examined the performance in terms of correct classification accuracy of each local descriptor within the BoF framework (Section 3.2.3). Finally, we investigated the effect of two different signatures on the BoF approach, described by Equation 3.7 and Equation 3.8.

4.3.1 Point Detector Evaluation

The objective was to evaluate which type of point detector is suitable for food classification, *i.e.* *DoG* that aims at locating stable points in scale-space representation to obtain self-similar points, or *entropy points* that aims at locating salient points based on the entropy measure. Similar to [179] we compared the two point detectors based on their classification performance. Since there is no formal approach

to solely compare point detectors for classification tasks without using any descriptor, we compare both point detectors using two of our descriptors namely *Steerable filters* and *SIFT* descriptors. For this evaluation, we used a subset of foods from the dataset described in 4.1.1 containing 37 different manually segmented food categories (Figure 4.1). For each food segmented region, *DoG* points and *entropy points* were detected. After the point was detected, *SIFT* features (Section 2.3.2), and *Steerable filters* which are 1st and 2nd statistic moments estimated from filtered images with 25 randomly oriented Gaussian filters (Section 2.3.2), were extracted over neighborhoods around each detected point. BoF with KNN was used for classification of these two local features. The *Steerable filter* feature vectors are 50 dimensional vectors, whereas the *SIFT* descriptors are 128 dimensional vectors. Table 4.3 shows the performance of *DoG* and entropy-based detector for food identification.

Table 4.3
Average classification rate of the *DoG* and *entropy points* detectors with *Steerable filters* and *SIFT* descriptors using BoF with KNN for classification. Experiments were performed with random selection of training and testing data, as well as increasing the percentage of training data (25%, 50%, and 75%).

**Point Detector + **	**Average Classif. Rate 25% training**	**Average Classif. Rate 50% training**	**Average Classif. Rate 75% training**
DoG + SIFT	0.58	0.66	0.78
Entropy points + SIFT	0.41	0.46	0.55
DoG + Steerable Filters	0.53	0.57	0.67
Entropy points + Steerable Filters	0.34	0.38	0.44

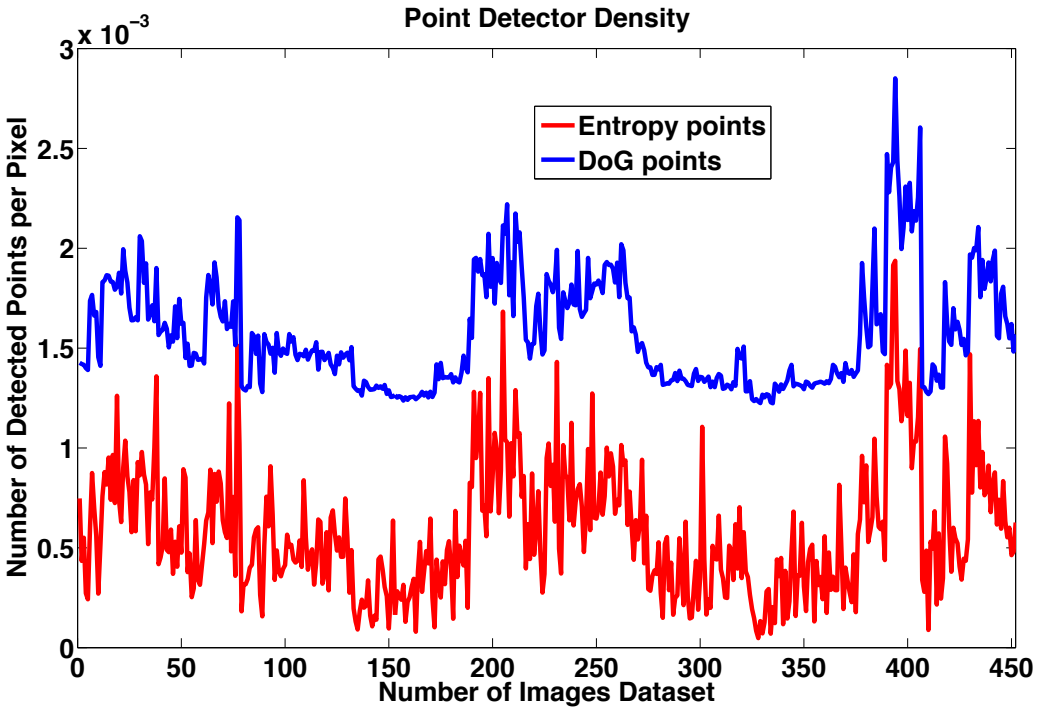


Fig. 4.5. Example Of Point Detector Density For All The Images In The Dataset: *DoG* (left), And *Entropy* Points (right). Point Detector Density Is The Ratio Between Number Of Points Detected And Size In Pixels Of The Segmented Region.

These results indicate performance similar to that presented in [179], where a point detector based on the Gaussian kernel achieved better performance than salience-based point detectors. For both descriptors *DoG* showed higher performance. A reason for that is that BoF model favors dense sampling strategies, *i.e.* the more points are detected (higher point density) the better object representation is accomplished. *DoG* was able to detect more than 50% more points for many food categories than *entropy points*. The point detector density per pixel can be seen in Figure 4.5, where point detector density is defined as the ratio between the number of points detected and size (in pixels) of the segmented region. *DoG* achieved more than double the point detection per pixel density compared to the *entropy*-based method.

Fig. 4.6. Example of point detectors: *DoG* (left), and *Entropy* points (right).

Another reason of obtaining better performance with *DoG* with respect to *entropy points*, was that many of the *entropy points* were detected around edges and corners, whereas *DoG* points were more evenly distributed. *DoG* points belonging to the edges are detected and eliminated beforehand as it has been described in Section 2.3.1. Figure 4.6 shows an example of points detected using both methods for an eating occasion image. The effect of having more points in the edge areas for the case of *entropy* points is very noticeable in Figure 4.6.

4.3.2 Local Descriptor Evaluation

We investigated local features in order to determine which ones captured more visual information for a large number of food classes (See Section 2.3.2). Table 4.4 shows the correct classification accuracy of the local features investigated, namely *SIFT* descriptor (dimension 128), *Red-SIFT* descriptor (dimension 128), *Green-SIFT* descriptor (dimension 128), *Blue-SIFT* descriptor (dimension 128), *SURF* descriptor (dimension 128), *TAMURA* local descriptor (dimension 3), *Steerable filters* (dimension 50), *DAISY* descriptor (dimension 200), *Gabor local* descriptor (dimension 48), *local color statistics* (dimension 20), *sum of MPEG-7 edge descriptors* (dimension 128), and *local-GOSDM* (dimension 60).

Table 4.4
Average classification rate of all local features using BoF model for classification and *DoG* as point detector. Two signatures are compared and classified using KNN. Experiments were performed with random selection of training data.

Local Descriptor	**Average (Std Deviation) KNN with** $Sign^{(1)}$	**Average (Std Deviation) KNN with** $Sign^{(2)}$
SIFT	0.59	0.65
R-SIFT	0.55	0.62
G-SIFT	0.54	0.61
B-SIFT	0.53	0.61
SURF	0.40	0.46
local TAMURA	0.36	0.43
Steerable Filters	0.36	0.43
DAISY	0.41	0.48
local Gabor	0.21	0.25
local Color Statistics	0.49	0.54
Sum of MPEG-7 Edge descriptor	0.40	0.46
local GOSDM	0.33	0.37

Within the BoF context we investigated two types of signatures (Section 3.2.3): signature $Sign^{(1)}$, with $Sig^{(1)} = \{\eta_1, ..., \eta_i, , .., \eta_N\}$ and η_i equal to the *term frequency-inverse document frequency*, and signature $Sign^{(2)}$, where $Sig^{(2)} = \{t_1, ..., t_i, .., t_N\}$ with t_i representing the frequency term (histogram count) of the ith cluster or visual word. In all cases the point detector was *DoG*.

The low performance of some features (*TAMURA local descriptor, local-GOSDM, Gabor local descriptor, local color statistics*) made us discard them for our final classification system (Figure 3.1). *local-GOSDM* proved to be not as efficient as the

GOSDM global version. In local neighborhoods gradient information is prone to noisy estimates, therefore modeling the gradient orientations was more successful in the global case.

There were other features such as *SIFT, SURF, DAISY*, or *Sum of MPEG-7 edge descriptors* that because of the way they are estimated they project similar feature spaces, and using all of them would increase the redundancy of our features. In order to determine which of these four descriptors contained more discrimination and descriptive information we examined the misclassifications of the best performing local feature, *i.e. SIFT*, and estimated the percentage of correct classification of the other three descriptors (*Sum of MPEG-7 edge descriptors, SURF, DAISY*). *Sum of MPEG-7 edge descriptors* and *SURF* obtained more than 10% of correct classification when *SIFT* failed, whereas *DAISY* obtained less 5% of correct classification when *SIFT* failed. Therefore we discarded *DAISY* for not complementing the SIFT descriptor. *SURF* and *Sum of MPEG-7 edge descriptors* describe the same information. Therefore *SIFT* and *SURF* were chosen for our final classification system.

Finally, among the local color features we selected the SIFT-based features (*Red-SIFT*, *Green-SIFT*, and *Blue-SIFT*) over the local color statistics due to their higher performance. From these experiments, we concluded that the following six features provided with the most complete visual representation of food categories, and thus were incorporated into our final classification system (Figure 3.1): *SIFT descriptor, Red-SIFT descriptor, Green-SIFT descriptor, Blue-SIFT descriptor, SURF*, and *Steerable filters*.

As far as the signature selection, the signature $Sign^{(2)}$ provided better results. This seems to indicate that for object classification tasks the *tf-idf* measure (Equation 3.7) is not a suitable signature representation in order to model visual vocabularies within the BoF approach. Simpler methods that use histogram counts of each visual word (Equation 3.8) can model better each visual word.

4.4 Overall Object Classification

The goal of this section is to present the evaluation of our multi-channel classification system for food classification (Figure 3.1). As we described above we discarded 5 local feature channels resulting in twelve feature channels that we used for our final overall system evaluation. These twelve feature channels (three color global feature channels: *global color statistics, entropy color statistics, predominant color statistics*, three texture global feature channels: *EFD, GFD, GOSDM*, and six local features channels: *SIFT descriptor, Red-SIFT descriptor, Green-SIFT descriptor, Blue-SIFT descriptor, SURF*, and *Steerable filters*) are individually classified by our classifier (SVM or KNN). The final class decision is obtained by combining the twelve individual decisions.

4.4.1 Datasets For Multi-Channel Classification Evaluation

In these experiments, we wanted to evaluate the performance of each individual classifier when automatic segmented regions were used in both training and testing. We investigated our proposed system on a food dataset using food items served in controlled and natural-eating-event (free living) studies and a well-known publicly available dataset (the PASCAL dataset) [180].

We used 303 eating occasion images containing 83 categories (79 foods, and utensils, glasses, plates, and plastic cups). Figure 4.7 shows examples of each food item. The number of segmented regions in each class varies from 15 to 50 per class.

The PASCAL Visual Object Classes (VOC) challenge dataset [180] is a benchmark in object recognition and detection, providing a standard dataset of images with annotated data. The PASCAL dataset contains intra-class variation in terms of object pose, orientation, size, illumination, and position. All the images are obtained from the Flickr photo-sharing website. The dataset contains 20 object classes, namely *Aeroplane, Bicycle, Bird, Boat, Bottle, Bus, Car, Cat, Chair, Cow, Dog, Horse, Motorbike, Person, Sheep, Sofa, Table, Potted Plant, Train*, and *TV/monitor*. Table

Fig. 4.7. Examples Of 83 Food Classes. (From left to right and top to bottom: *Apple, Apple juice, Bagel, Banana, BBQ Chicken, Broccoli, Brownie, Carrot, Celery, Cheese Burger, Chicken Wrap, Chocolate Cake, Chocolate Chip Cookie, Clementine, Coffee, Coke, Cream Cheese, Cup, Egg (Scrambled), English Muffin, French Dressing, French Fries, Frozen Meat Loaf, Frozen Meal Turkey, Fruit Cocktail, Garlic Bread, Glass, Goldfish, Granola Bar, Grapes, Gravy Chicken, Green Beans, Ham Sandwich, Ice Cream, Jelly, Ketchup, Lasagna, Lettuce, Mac and Cheese, Margarine, Mashed Potatoes, Mayo, Milk, Muffin, Non Fat Dressing, Noodle Soup, Orange, Orange Juice, Pancake, Peach, Peanut Butter, Pear, Peas, Pineapple, Pizza, Plate, Pork Chop, Potato Chips, Pretzel, Pudding, Ranch Dressing, Rice Krispy Bar, Salad Mix, Saltines, Sausages, Snicker Doodle, Snicker, Spaghetti, Strawberry, Strawberry Jam, String Cheese, Sugar Cookie, Syrup, Utensil, Vegetable Soup, Watermelon, Wheat Bread, Wheaties, White Toast, Yellow Cake, and Yogurt.*

Table 4.5
Queries used to retrieve images from Flickr. Words in bold show the "targeted" class 4.5.

Category	Queries
Aeroplane	airplane, plane, biplane, monoplane, aviator, bomber, hydroplane, airliner, aircraft, fighter, airport, hangar, jet, boeing, fuselage, wing, propellor, flying
Bicycle	bike, cycle, cyclist, pedal, tandem, saddle, wheel, cycling, ride, wheelie
Bird	birdie, birdwatching, nest, sea, aviary, birdcage, bird feeder, bird table
Boat	ship, barge, ferry, canoe, boating, craft, liner, cruise, sailing, rowing, watercraft, regatta, racing, marina, beach, water, canal, river, stream, lake, yacht
Bottle	cork, wine, beer, champagne, ketchup, squash, soda, coke, lemonade, dinner, lunch, breakfast
Bus	omnibus, coach, shuttle, jitney, double-decker, motorbus, school bus, depot, terminal, station, terminus, passenger, route
Car	automobile, cruiser, motorcar, vehicle, hatchback, saloon, convertible, limousine, motor, race, traffic, trip, rally, city, street, road, lane, village, town, centre, shopping, downtown, suburban
Cat	feline, pussy, mew, kitten, tabby, tortoiseshell, ginger, stray
Chair	seat, rocker, rocking, deck, swivel, camp, chaise, office, studio, armchair, recliner, sitting, lounge, living room, sitting room
Cow	beef, heifer, moo, dairy, milk, milking, farm
Dog	hound, bark, kennel, heel, bitch, canine, puppy, hunter, collar, leash
Horse	gallop, jump, buck, equine, foal, cavalry, saddle, canter, buggy, mare, neigh, dressage, trial, racehorse, steeplechase, thoroughbred, cart, equestrian, paddock, stable, farrier
Motorbike	motorcycle, minibike, moped, dirt, pillion, biker, trials, motorcycling, motorcyclist, engine, motocross, scramble, sidecar, scooter, trail
Person	people, family, father, mother, brother, sister, aunt, uncle, grandmother, grandma, grandfather, grandpa, grandson, granddaughter, niece, nephew, cousin
Sheep	ram, fold, fleece, shear, baa, bleat, lamb, ewe, wool, flock
Sofa	chesterfield, settee, divan, couch, bolster
Table	dining, cafe, restaurant, kitchen, banquet, party, meal
Potted Plant	pot plant, plant, patio, windowsill, window sill, yard, greenhouse, glass house, basket, cutting, pot, cooking, grow
Train	express, locomotive, freight, commuter, platform, subway, underground, steam, railway, railroad, rail, tube, underground, track, carriage, coach, metro, sleeper, railcar, buffet, cabin, level crossing
TV/Monitor	television, plasma, flatscreen, flat screen, lcd, crt, watching, dvd, desktop, computer, computer monitor, PC, console, game

4.5 illustrates the text queries used to retrieve the images from Flickr. Figure 4.8 shows an example for each object class. From the PASCAL dataset we randomly selected 35 images from each of the 20 categories from the 2009 and 2010 datasets.

Fig. 4.8. Examples Of The 20 Categories In The PASCAL Database. (From left to right and top to bottom: *Aeroplane, Bicycle, Bird, Boat, Bottle, Bus, Car, Cat, Chair, Cow, Dog, Horse, Motorbike, Person, Sheep, Sofa, Table, Potted Plant, Train*, and *TV/monitor*)

4.4.2 Performance Evaluation Of Each Individual Channel

So far we have described a set of experiments to determine the best color, texture and local feature channels in terms of classification accuracy for food classification. Based on their performance, twelve feature channels are used for our final multichannel classification system. The final set of feature spaces we used included three global color features, namely *Global color statistics, Entropy color statistics*, and *Predominant color statistics*, three global texture features, namely *GOSDM, EFD*, and *GFD*, and finally, six local features *SIFT descriptor, Red-SIFT descriptor, Green-SIFT descriptor, Blue-SIFT descriptor, SURF/sum of MPEG-7 edge descriptor*, and *Steerable filters*. We individually classified each feature channels using either SVM or KNN and performed a late decision fusion on the outcomes of each individual classifier. The outcome from the classifier includes the decision label, the confidence score, and top β candidate decisions of each feature channel.

First, we experimentally compared the visual information captured by each feature channel. Table 4.6 shows mean classification rate for each feature channel for both the Food and PASCAL datasets and for both classifiers: KNN and SVM. In the experiments 60% of the data are used for training/validation and 40% for testing. Experiments were performed with random selection of training and testing data. Note that in the SVM case, *global color statistics, entropy color statistics, EFD, SIFT, Red-SIFT, Green-SIFT, Blue-SIFT,* and *Steerable filters, GOSDMs* where classified using RBF kernels, and for *predominant color statistics, GFD* and *SURF/Sum of MPEG-7 Edge descriptors* used quadratic kernels. As shown, the *global color statistics* feature outperformed the other color features in all tests. Among texture features *EFD* performed better than the other two types of features: *GFD*, and *GOSDM*. Also, local features and the BoF approach proved to be very efficient. These results show the importance of color information to globally describe food segmented regions. Finally, local features such as SIFT achieved high performance, but so did *Red-SIFT*, *Green-SIFT*, and *Blue-SIFT* which indicates that color information may also be relevant to

describe objects locally. From these results, we also observed that, in general, for local features SVM performs better than KNN on the individual feature channels. This confirms one of the statements made in Section 3.2.1, where KNN is susceptible to the curse of dimensionality. Local feature vectors have, in general, higher dimension than the global feature vectors.

Table 4.6
Mean classification rate for all classes for each type of feature channel using a KNN and SVM classifiers. L indicates local feature. G indicates global feature.

Feature Channel	**Mean classif. rate KNN Food Dataset**	**Mean classif. rate SVM Food Dataset**	**Mean classif. rate KNN PASCAL Dataset**	**Mean classif. rate SVM PASCAL Dataset**
Color Stats. (G)	0.68	0.62	0.48	0.51
Entropy Color Stats. (G)	0.20	0.35	0.38	0.41
Pred. Color Stats. (G)	0.42	0.60	0.35	0.54
EFD (G)	0.39	0.47	0.36	0.40
GFD (G)	0.23	0.27	0.26	0.33
GOSDM (G)	0.32	0.32	0.32	0.36
SIFT (L)	0.44	0.48	0.49	0.59
Red-SIFT (L)	0.45	0.48	0.46	0.56
Green-SIFT (L)	0.44	0.49	0.44	0.54
Blue-SIFT (L)	0.47	0.47	0.45	0.54
SURF/Sum of MPEG-7 Edge (L)	0.43	0.45	0.38	0.43
Steerable Filters (L)	0.39	0.43	0.32	0.41

Table 4.7
Average classification rate for each decision fusion approach Majority vote rule and Maximum confidence score for both KNN and SVM classifiers for multiple candidates (1, and 8)

Decision Fusion (Classifier)	**1 Candidate**	**8 Candidates**
Majority vote rule (KNN)	0.70	0.74
Maximum confidence score (KNN)	<0.1	0.75
Majority vote rule (SVM)	0.57	0.72
Maximum confidence score (SVM)	0.33	0.70

4.4.3 Decision Fusion Evaluation

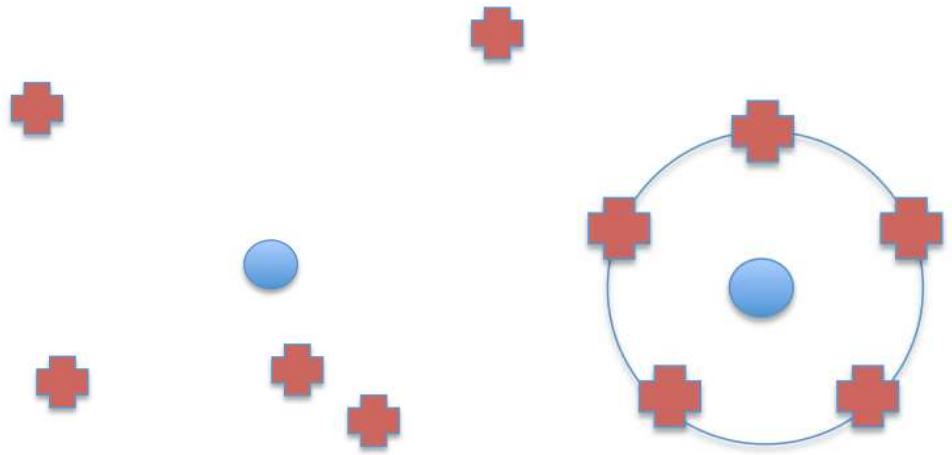
In this section, we compared the performance of both decision fusion approaches described earlier to combine the feature channels. The experiment consisted of comparing our two decision fusion methods as a function of the number of candidate classes (1 and 8) on each feature channel. By using more than one candidate for each channel, the probability that the true (correct) label be among the class candidates increases. Table 4.7 shows the performance of both decision fusion approaches as the number of candidates is increased.

When the number of candidates is one, only the majority vote with KNN outperforms the best individual feature channel (*Global color statistics* as shown in Table 4.6), which indicates that maximum confidence score decisions are of interest only when using multiple candidates for each feature channel.

As the number of candidates increases, the maximum confidence score achieves classification rates similar to the majority rule for both KNN and SVM. However, for small number of candidates using KNN, the performance of the maximum confidence

score is very low compared to SVM with maximum confidence score. This can be related to the scoring system of each classifier. The KNN confidence score only depends on the location of the nearest neighbors, while in the SVM case the score is a function of all the class segmented regions.

In the KNN case, majority rule provided higher classification rates than the maximum confidence score for small number of candidates. One of the reasons is due to the treatment that each neighbor receives from the decision fusion strategy, *i.e.* majority rule assumes that a class receives maximum score if it is selected by majority vote regardless of the spatial location of the neighbors, while maximum confidence score assigns scores based on the spatial location of the neighbors. This effect distorts the spatial distribution of training data points by assuming that the k nearest neighbors are equally far from the testing data, while the maximum confidence score uses the real spatial distribution of the neighbors to obtain the final decision. Figure 4.9 illustrates the effect of majority vote rule on the k nearest neighbors over a testing segmented region.



(a) Confidence Score Rule Effect Over k nearest neighbors

(b) Majority Vote Rule Effect Over k nearest neighbors

Fig. 4.9. The Effect Over $k = 5$ Nearest Neighbors By (a) Max Confidence Score (Real configuration) And (b) Majority Vote Rule Criteria.

In the SVM case, the maximum confidence score and majority rule obtained similar performance for larger number of candidates, which indicates that the scoring system proposed for SVM does not depend on the spatial distribution of k nearest neighbors, but in the feature space of the entire class.

Majority rule penalizes misclassifications more than maximum confidence score. Imagine the following situation, where the true label appears as top candidate in 7 out of the 12 feature channels, whereas an incorrect class appears in 9 out of the 12 feature channels but as second or even third candidate. In this situation majority rule will choose the incorrect class label, whereas maximum confidence score will select the true label if the sum of confidence scores is larger than the sum of confidence scores of the incorrect class.

One of the conclusions of these results is that decision fusion strategies require very distinct (orthogonal) feature spaces, *i.e.* feature channels that complement each other. The maximum confidence score has a great dependance upon the training data since it bases its efficiency on the visual appearance compactness of a class to assign scores. If unstable segmented regions within each class have large variation in appearance, majority rule proved to be a better solution. Another observation from these experiments was that locality-based classifiers such as KNN have more consistent class decision, even for very independent feature spaces like ours.

We were also interested on determining the contribution of each feature channel in the final food classification decision. Table 4.8 illustrates the percentage of agreement of each channel with the final decision. SIFT-based features contributed the most in the final decision as far as local features. EFD, texture feature, contributed in some cases more than some color features, which indicates that texture features contain valuable information for food characterization (Table 4.8). Local features are important in the final decision for food items composed of many ingredients such as *cheeseburger, soups, sandwiches*, whereas both types of global, color and texture, features describe more uniform characteristics. In general, texture features do not capture as much information as color features, and are more susceptible to blurring

artifacts. But these proposed texture features are useful for food classification when decisions based on color information are incorrect.

Table 4.8
Percentage of agreement of each feature channel classifier with the final decision after fusion using 1, 4, and 8 candidates. Note these are percentage averaged for all classes in the food database

Feature Channel	**Decision Agreement Percentage 1 Candidate**	**Decision Agreement Percentage 4 Candidates**	**Decision Agreement Percentage 8 Candidates**
Color Stats. (G)	61.7%	76.0%	79.6%
Entropy Color Stats. (G)	23.8%	48.5%	59.8%
Pred. Color Stats. (G)	42.5%	65.0%	70.3%
EFD (G)	41.9%	64.8%	69.8%
GFD (G)	23.0%	44.2%	55.2%
GOSDM (G)	33.6%	56.7%	67.4%
SIFT (L)	55.3%	79.4%	85.9%
Red-SIFT (L)	61.6%	83.5%	89.5%
Green-SIFT (L)	61.5%	83.0%	88.8%
Blue-SIFT (L)	60.8%	83.1%	90.3%
SURF (L)	52.5%	75.1%	78.9%
Steerable Filters (L)	46.3%	72.0%	80.6%

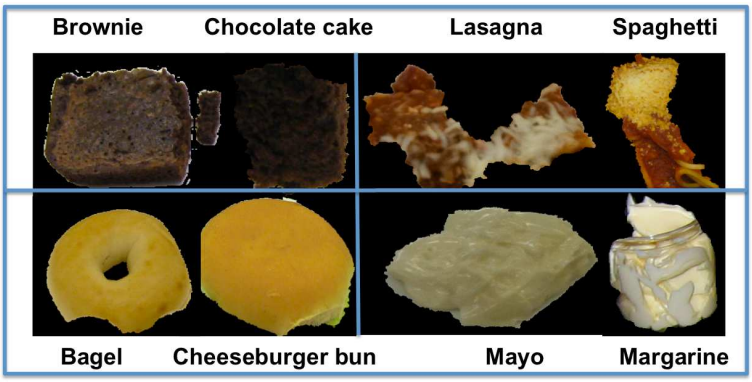


Fig. 4.10. Examples Of Misclassified Food Segmented Regions.

4.4.4 Codebook Refinement

We were also interested in comparing the contribution of introducing a codebook refinement step in the BoF model, by either using PCA in the local feature spaces before the word formation or considering KECA clustering as opposed to hierarchical k-means.

In both cases PCA and KECA did not show any improvement in terms of correct classification rate in the local feature channels. In particular, when using KECA clustering instead of hierarchical k-means, the results were considerably worse. The reason can be seen in Figures 4.11.a and 4.11.b, where it shows the distance among the SIFT signatures for a subset of the foods. Because of the way KECA clusters feature vectors, the distance between signatures increased more than 14% on average, which indicates that KECA performs sparser data projection.

By doing PCA before the word formation using hierarchical k-means, we did not observe any improvement in terms of correct classification rate. One reason for such results is that the effect of PCA is not designed to retain class discriminant information, but rather to reduce the feature space dimensionality based on correlation structure. Thus, PCA failed at capturing discriminant features while reducing dimension.

4.4.5 System Performance With Contextual Information

The goal of incorporating refinement by means of contextual information is to correct potential misclassified segmented regions based on classifier confidence and the sequence of most likely labels provided by the CRF. In some cases, we have observed that the contextual information refinement step is able to correct up to **10%** of the segmented regions in the image. Figure 4.12 shows an example of misclassified segmented regions before context refinement (left), and corrected misclassifications by using context (right). Foods such as *fruit cocktail, ketchup and coffee* were corrected

to *pinneapple, french dressing, coke* respectively by using the contextual information of the image.

One of the disadvantages of CRF to model contextual information is the requirement of large annotated training data in order to correctly model object interactions (object occurrences in the scene). From our experiments, we have observed that CRF is prone to over-fitting, especially to noisy data or to small training datasets. This aligns with the observations of Gregory et. al. in the context of language learning [181].

Some objects were, and always will be, inherently difficult to classify due to their similarity in any feature spaces we could consider. Food items are no exception. Figure 4.10 shows examples of the segmented regions of some of these categories and corresponding misclassifications. Some of the misclassifications can be corrected by considering the contextual information, *margarine* and *mayo*, or *bagel* and *cheeseburger bun*, or *ketchup* and *catalina dressing*. However, other examples such as *spaghetti* and *lasagna* or *chocolate cake* and *brownie* cannot be successfully modeled by this type of contextual information. One way to solve this can be accomplished by examining individual eating patterns (contextual information) to predict the likelihood of eating certain foods (e.g. *chocolate cake* vs. *brownie*) and improve the precision of the classifier.

4.5 System Evaluation: User Study

For this final set of experiments, we evaluated our food identification system for real life scenarios. We conducted several user studies under controlled and free-living conditions. Controlled user studies took place at earlier stages of our system development. The goal was to constrain our problem by controlling lighting conditions, and background. In controlled studies, we provided the participants with all the foods and beverages, as well as plates, glasses, and silverware. Free-living studies consisted of participants acquiring images of their eating occasions under real life situations (e.g.

eating at home watching TV, eating at the office, eating on-the-go). Although most foods were provided; silverware, plates, beverages were not provided to the users. Lighting and background conditions were not controlled.

4.5.1 Classification Results: Controlled User Study

As mentioned earlier, in our controlled user studies, the food was provided to users. A total of 28 food items were provided in several meal sessions. Lighting and background conditions were known. Users attended to our campus for these meal sessions where they acquired the images. Figure 4.13 shows some images acquired by participants under controlled conditions.

Figure 4.14 shows the confusion matrix for the 28 food items. Segmentation for these experiments was performed manually (hand segmentation). The average correct classification rate was **0.986**. These results show our system's performance in ideal conditions (groundtruth data, controlled lighting and background).

4.5.2 Classification Results : Free-living User Study

Figure 4.15 shows some images acquired by participants in real life conditions. The experiment consisted of collecting images of 44 participants for a 7-day period. Each participant was asked to acquire a pair of before and after eating occasion images at each eating occasion for a week and send them to our servers. These images were automatically processed by our system. A total of 1562 before images were processed, the most number of images acquired by a participant were 88 and the least number of images were 23. Finally, the average number of before images acquired per week/per participant was 37.

The training data for these experiments was obtained beforehand by acquiring images in a series of food imaging sessions. The goal was to obtain training data of the foods that participants would take home in order to create a learning model so that the classifier could perform food identification on the images sent by the users.

The outcome was measured using the identification rate defined as:

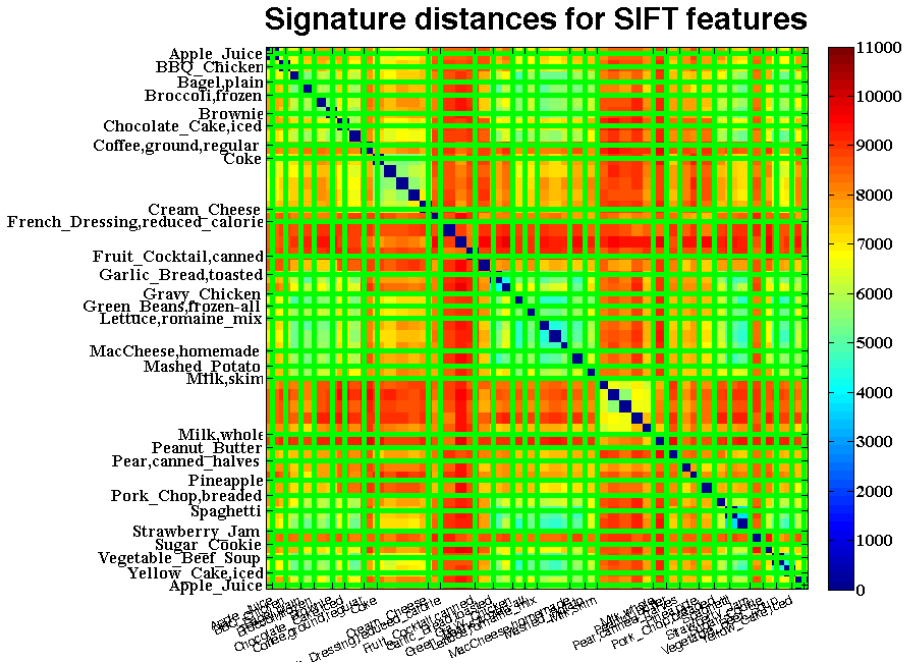
$$Rate = \frac{TP}{TP + FP/k + TN} \tag{4.3}$$

where TP indicates *True Positives* (correctly detected segmented regions), FP indicates *False Positives* (Food missed identified). TN indicates *True Negatives* (Food not detected by the segmentation step). Finally, k refers to the identification rate order. If we are interested in knowing the identification rate using the top 4 outputs (food labels) of the classifier, then we set $k = 4$. Note that when $k = 1$, the identification rate measure is equivalent to classification accuracy [182]. Table 4.9 shows the the identification rate using the top 1, and 4 food category outputs from the classifier. Figures 4.16.a and 4.16.b shows the identification rate for each participant using

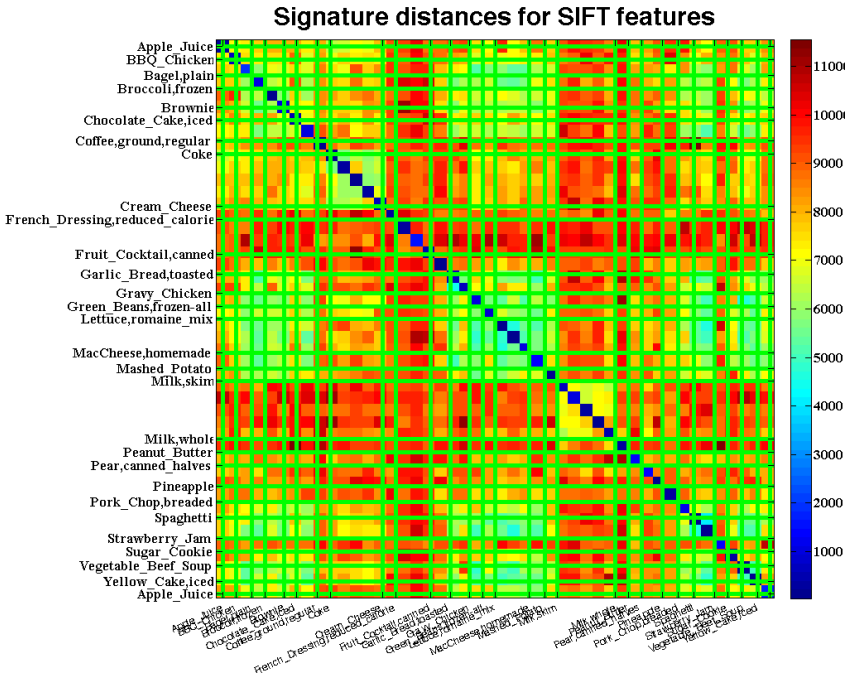
Table 4.9
Identification Rate using the top 1, and 4 class suggestions given by the classifier.

	Top 1	**Top 4**
Identification Rate	0.43	0.74

top 1 food category output and top 4 food category outputs respectively. The identification rate differences between participants are due to lighting conditions across images acquired by different participants (Figures 4.17.1, 4.17.2, 4.17.3, Figures 4.17.4 and 4.17.6), cluttered background (Figure 4.17.1, Figure 4.17.4), different food setups (colored plates or no plates at all) with respect to our training datasets which caused unstable segmentations (Figure 4.17.7, Figure 4.17.8, and Figure 4.17.9), and blurring artifacts that caused food items to be mis detected (Figure 4.17.3 and Figure 4.17.5). Accounting for these image distortions and effects is the next challenge in our image analysis system. We believe that through these sets of experiments we have found the most common problems that we can encounter in free-living situations.



(a) Signature Distance For SIFT Features Using Hierarchical K-means



(b) Signature Distance For SIFT Features Using KECA Clustering

Fig. 4.11. Signature Distances For SIFT Features Using (a) Hierarchical K-means And (b) KECA Clustering.

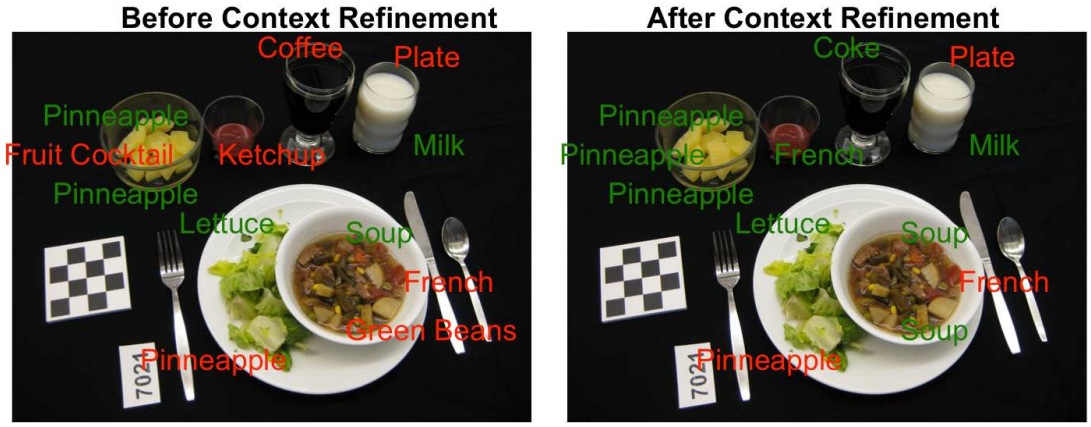


Fig. 4.12. An Example Of Misclassified Segmented Regions After Context Refinement (Left Image), And Corrected Misclassifications By Contextual Information (Right Image). Red Labels Represent Misclassifications And Green Labels Represent Correctly Labeled Segmented Regions.

Fig. 4.13. Examples of Meal Images In A Controlled User Study.

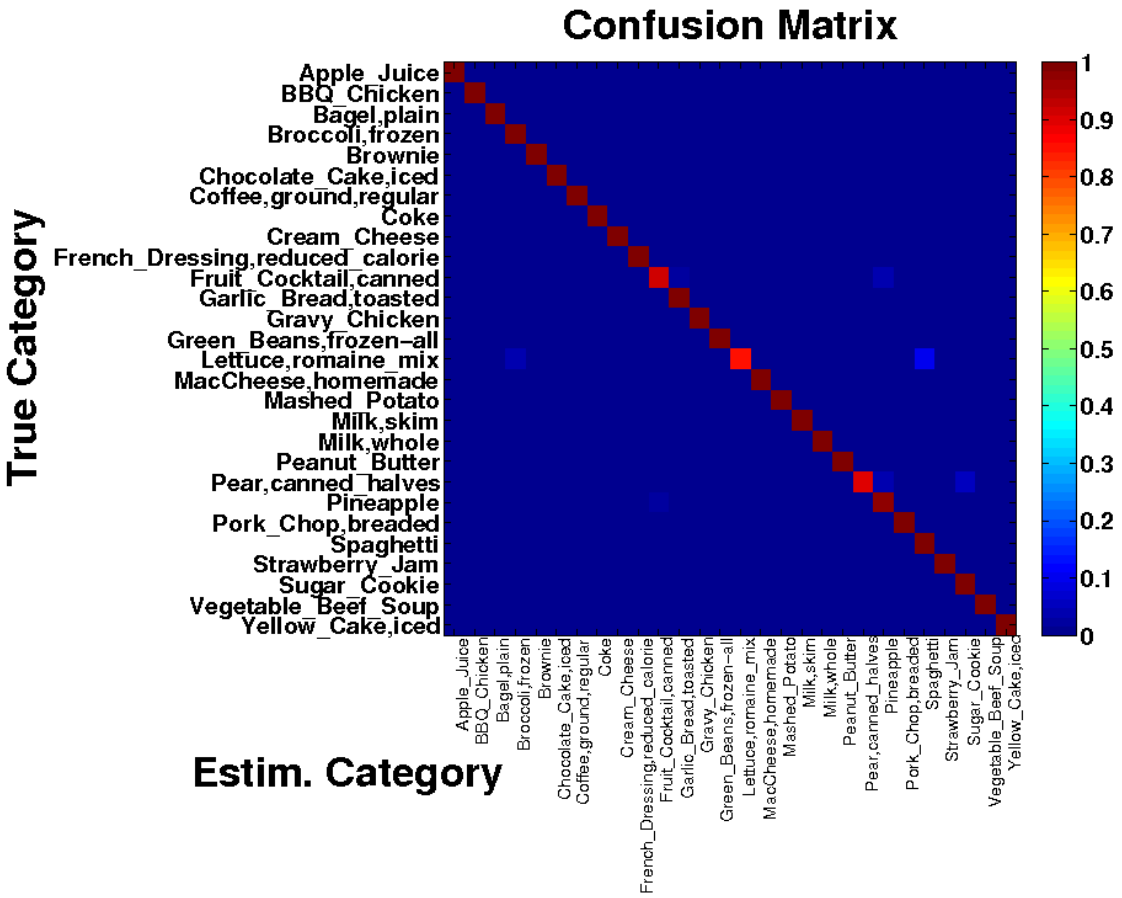
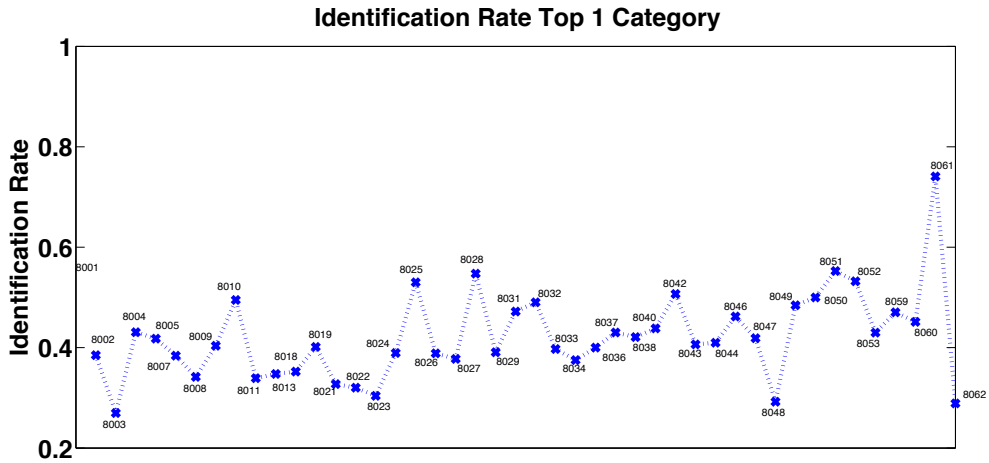
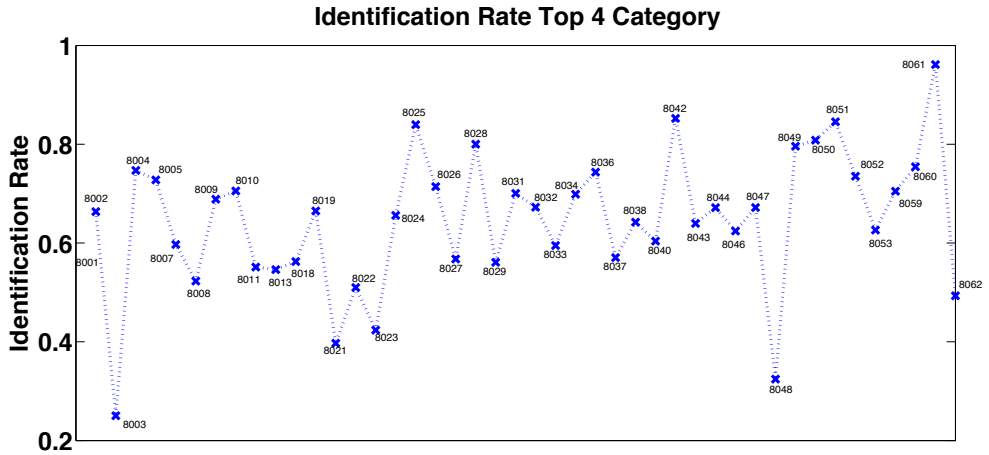


Fig. 4.14. Confusion Matrix For Controlled User Studies. 28 Food Classes. Groundtruth Segmentation.

Fig. 4.15. Examples of Eating Occasion Images Acquired In Free-living User Study.



(a) Identification Rate Top 1 Choice



(b) Identification Rate Top 4 Choice

Fig. 4.16. Identification Rate for Each Participant using (a) Top 1 and (b) Top 4 Food Classes as Classifier Output.

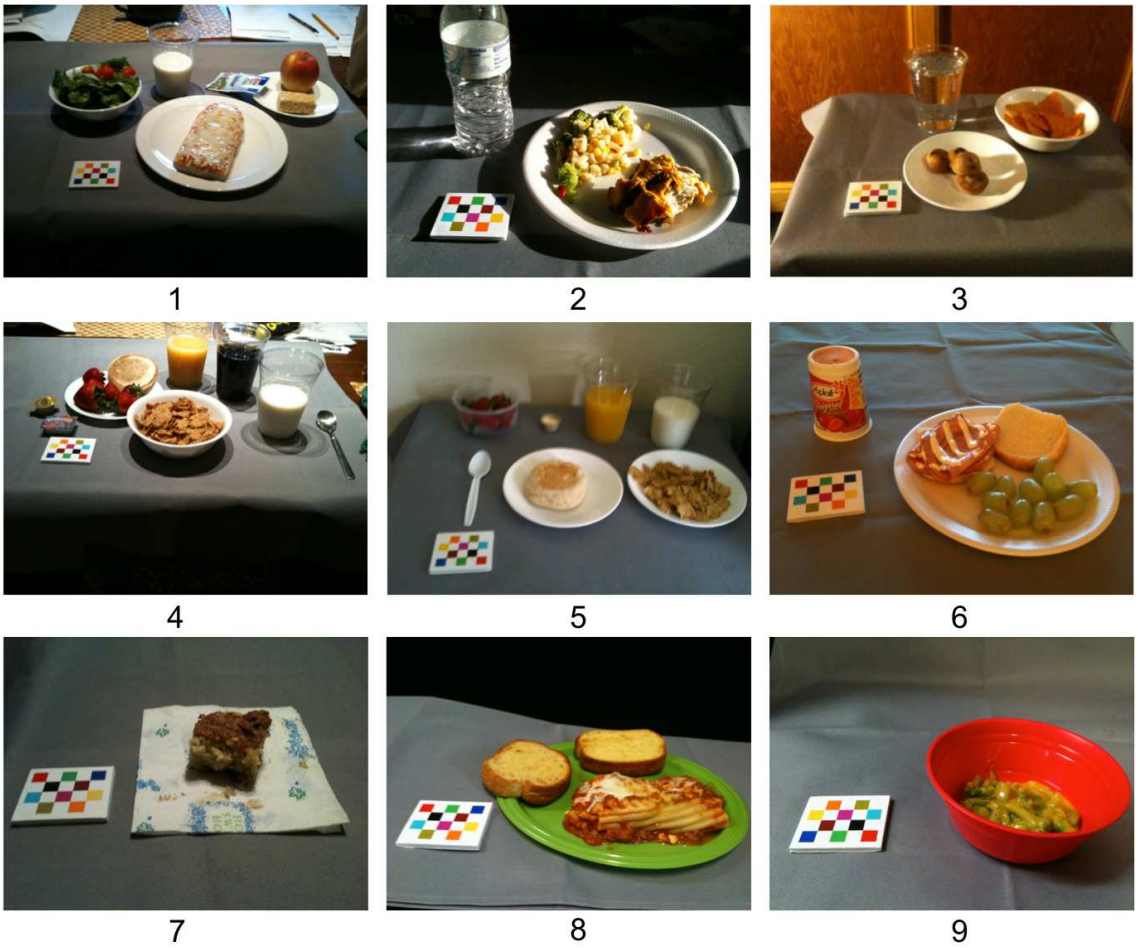


Fig. 4.17. Examples Of Difficult Images To Classify Due To Different Lighting Conditions (1, 2, 3 , 4, and 6), Cluttered Background (1, 4), Blurred Images (3, 5), And Different Food Setups - Colored plates Or No Plates At All - (7, 8, and 9).

5. INTEGRATED IMAGE DATABASES FOR DIETARY ASSESSMENT

5.1 TADA Databases

One of the goals of the Technology Assisted Dietary Assessment (TADA) project is to provide tools to dietitians, nutritionists and researchers in order to determine dietary habits and patterns of individuals or entire communities. Data mining techniques are often used for such tasks with the data stored in Database Management Systems (DBMs). DMBs and their accompanying data communications systems have become the foundation for building core applications in many problem domains. They provide an efficient way to investigate complex applications without having to duplicate data access and retrieval functions for each application [183]. DMBs also provide a simple way to share data among multiple applications and users.

As part of the TADA system, a DMB system has been created in order to manage and store the data generated by the TADA system. We have organized the data around three key elements: images, foods, and users. Based on these elements our database system is composed of three logically interrelated databases namely *I-TADA, T-FNDDS*, and *E-TADA*. The *I-TADA* database contains information related to the food image. The *T-FNDDS* database is an extension of the original USDA Food and Nutrient Database for Dietary Surveys (FNDDS) [76] by adding visual descriptions that can be used for image analysis and other information associated with each food item such as barcode data. Finally, the *E-TADA* database stores information available

for each user and data related to our user nutrition studies [1]. In this section, we describe each of the three databases in detail as well as their content.

Figure 5.1 shows the main information stored in each of the three databases.

I-TADA	E-TADA	T-FNDDS
• Original Image • Ground truth segmented images – used to validate image analysis • Results of the automatic food classification methods • "Before meal" and "after meal" pair images with time difference • Automatic segmented images from image analysis • List of foods in the image • Exif data from the camera • Geocoordinates where the image was acquire • Connections to E-TADA and T-FNDDS	• Complete menus served in the nutritional studies • Date(s) of the study • Participant's information: visit date, participant ID, date of birth, age, gender, height, weight, ethnicity, race and zip code • Automatic image analysis results • Participant's statistics: frequently eaten foods, meal average time • Connections to the I-TADA and T-FNDDS	• Visual and nutrition description of each food • Barcode information • FNDDS (Food and Nutrient Database for Dietary Assessment) database • Food densities • Connections to E-TADA and I-TADA

Fig. 5.1. The Main Components And Contents For Each TADA Database.

[1]The *E-TADA* system does not contain any information that allows a user of our system to be individually identified. The TADA system is compliant with all US regulations with respect to user privacy. A health care professional could use our system to aid in the treatment of a particular user by relating their TADA User ID to their actual identity, however this information is not in any of our databases and was not available to us in any of our user studies.

5.1.1 I-TADA

The *I-TADA* database can be considered as a food image database that contains data generated from the food image. As soon as an image is available at the server, the server builds a data structure with the information that can be obtained from the image. In general, images generated from digital cameras contain data integrated into the image file header. This is known as *image metadata*. Most of it comes from the EXIF (Exchangeable Image File Format) header of the image file [184]. This includes date and time stamps, camera specifications such as camera model or make, internal parameters associated to each camera/lens such as aperture, focal length, shutter speed. An example of a set of EXIF data from different images is shown in Figure 5.3. When available, geolocation information can also be obtained (e.g. GPS data). The EXIF data is used to evaluate the image analysis performance across different camera models and other EXIF parameters. Also, the camera internal parameters are used for camera calibration. The GPS information and the GPS timestamp is incorporated into *I-TADA*. This information is used in the discovery and analysis of eating patterns and to aid the image analysis.

The information available for each image in *I-TADA* includes the list of all the food items in the image from the automatic image analysis and the results confirmed by the user. For each original food image the following is available: its corresponding before/after eating occasion image, the time difference between both images, the user who acquired the images (numeric user ID only), a tag indicating which study they participated in, files associated to the image such as groundtruth data (manually segmentations) and automatic segmentation results, EXIF data, date the image was acquired, and if available, geocoordinates where the image was acquired, as well as connections to *E-TADA* and *T-FNDDS*. Figure 5.2 shows an example of the data available.

To assess the accuracy of our image analysis approaches, it is important to develop groundtruth data for the images. We manually extract each food item in an image

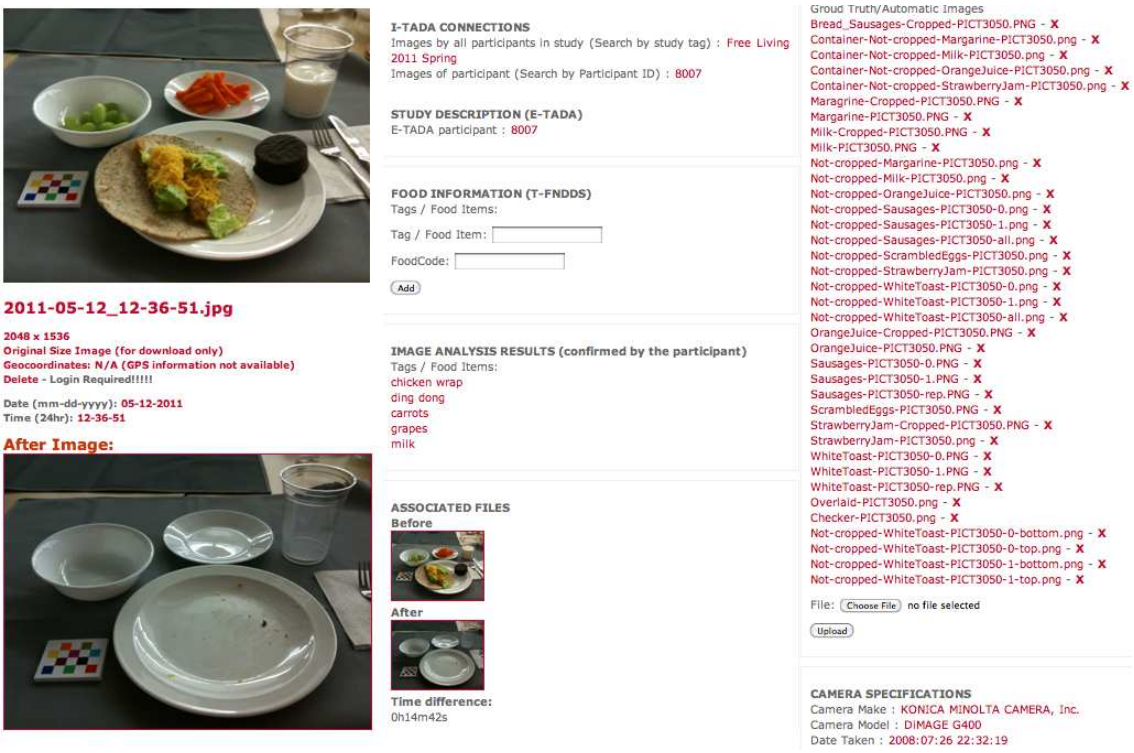


Fig. 5.2. An Example Of The Information Available In I-TADA For A Particular Image.

using a Cintiq Interactive Pen LCD Display and Adobe Photoshop. Given a food image, we trace the contour of each food item and generate corresponding mask images along with the correct food labels. All groundtruth data is stored in *I-TADA*.

In the TADA system, EXIF data is used to evaluate the image analysis performance across different camera models and other EXIF parameters. Also, camera internal parameters are used for camera calibration (Figure 5.3).

As mentioned before, the GPS information including the geolocation coordinates where the image was acquired, and the GPS timestamp is also incorporated into *I-TADA*. The reasons we use metadata (e.g. EXIF or GPS data) in our system are twofold. First, as an "extra feature" by providing a rich set of information to aid the image analysis. Secondly to obtain patterns of dietary habits [185]. GPS information can be used to identify foods. For example, from images that children acquired at

CAMERA SPECIFICATIONS	CAMERA SPECIFICATIONS	CAMERA SPECIFICATIONS
Camera Make : Canon	Aperture : 4281/1441	Camera Make : Panasonic
Camera Model : Canon PowerShot SD200	Color Space: sRGB	Camera Model : DMC-FZ4
Date Taken : 2008:08:07 04:21:29	Exposure Mode: Auto Exposure	Date Taken : 2008:09:14 14:24:03
Aperture : 4.96875 (F5.6)	FNumber : 14/5	Aperture : unknown
ISO Equivalent: Auto	Flash : Not Available	ISO Equivalent: 200
Endian : II	Metering Mode : Average	Endian : II
Focal Length: 5.80 mm	White Balance : Auto	Focal Length: 6.00 mm
Flash : Fired - Red Eye, Auto-Mode	GPS Time Stamp (UTC) : 18:54:42.8	Flash : Off
Software : None	Camera Make : Apple	Software : Ver.1.0
Mettering Mode : pattern	Camera Model : iPhone 3G	Mettering Mode : pattern
Brightness : None	Camera Orientation : Horizontal (normal)	Brightness : None
Exposure : 1/60 sec, program: unknown	Device OS version : 3.1.3	Exposure : 1/30 sec, program: normal

Fig. 5.3. Examples of EXIF data available for three digital cameras.

school we can use the geolocation and the timestamp information to determine what food was served since many schools post their menus in advance and this information can be incorporated into our database. Another situation is when the user eats certain foods at specific locations and times. If a user eats the same food on a particular day of the week at a given location, the GPS information and time stamp can be used to constrain the food candidates in the classification process, and thus, improve accuracy.

In *I-TADA*, the foods eaten by a user are stored so that the food identification system can potentially use this information to create "individual" sets of training data related to each user in order to learn eating patterns [185]. We can use *Google Maps* to retrieve restaurant information (menus) from restaurants that are located where the image was acquired and use this information as "side information" in the image analysis. The timestamp data can be used to estimate the time elapsed between the beginning and end of an eating occasion. In the current system, we use this information to program "reminder functions" on the mobile telephone so that the user is prompted to acquire the "after" eating occasion image after finishing his/her eating occasion.

All the data available in *I-TADA* is interconnected to data available in the *T-FNDDS* and *E-TADA*.

5.1.2 T-FNDDS

The goal of the *T-FNDDS* is to extend an existing food database with the types of information needed for dietary assessment from the analysis of food images and other metadata. Our proposed database is an extension of the USDA FNDDS [76]. The FNDDS is a public database of foods, their nutrient values, and weights for typical food portions. FNDDS has 11 tables including primary descriptions for approximately 7,000 food items, weights for various portions of each food (approximately 30,000 weights), complete nutrient profiles (food energy and 60 nutrient/food components), descriptions and measurement units for each nutrient, and descriptions for approximately 6,500 similar food items associated with specific main food items. Each food is described by an 8-digit unique food code. We have imported the original FNDDS into the *T-FNDDS*. An example of a typical entry is shown in Figure 5.4. As a result of our approach of using food images to estimate the nutrients consumed and energy intake of an eating occasion, we developed many visual characteristics (features) that are used for each food item so that foods can correctly be identified from their images. These features describe elementary characteristics such as color and texture. These features can be grouped together forming a collection of metadata that we call the Visual Characterization Metric (VCM). The VCM is a unique way for food to be indexed in the food database and provide the research community with features needed to recognize food items from food images. Our goal is to complement a classical nutrient database with emerging methods of dietary assessment such as food imaging.

We are currently developing the basic data structure of the VCMs using features that are stored in our database as a result of the image analysis. These include color, perceptual features, type index (solid/liquid), reflectivity, among others. Figure 5.6 illustrates the VCM concept for a eating occasion image. For each of the food items identified in the image, a set of visual descriptors are constructed. A standard reference language must be used to describe the VCM. This can be accomplished by using

Egg omelet or scrambled egg, NS as to fat added in cooking

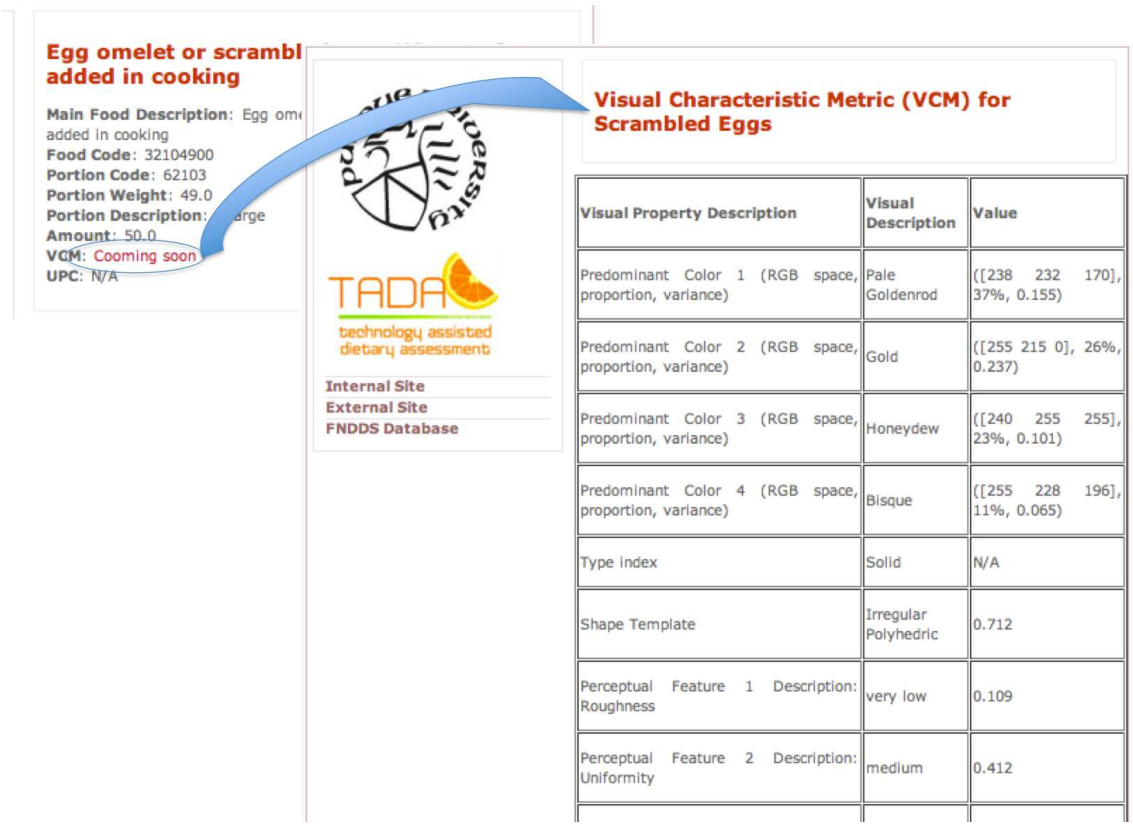
Main Food Description: Egg omelet or scrambled egg, NS as to fat added in cooking
Food Code: 32104900
Portion Code: 62103
Portion Weight: 49.0
Portion Description: 1 large
Amount: 50.0
VCM: Cooming soon
UPC: N/A

Nutrient Description	Nutrient Value per 100g	Nutrient Code
Protein	10.98	203
Total Fat	11.59	204
Carbohydrate	1.81	205
Energy	155.0	208
Alcohol	0.0	221
Water	74.02	255
Caffeine	0.0	262
Theobromine	0.0	263
Sugars, total	1.89	269
Fiber, total dietary	0.0	291
Calcium	72.0	301
Iron	1.49	303
Magnesium	12.0	304
Phosphorus	178.0	305
Potassium	146.0	306
Sodium	410.0	307
Zinc	1.0	309
Copper	0.086	312
Selenium	26.4	317
Vitamin A, RAE	156.0	320
Carotene, beta	32.0	321
Carotene, alpha	0.0	322
Vitamin E (alpha-tocopherol)	1.13	323
Cryptoxanthin, beta	7.0	334
Lycopene	0.0	337
Lutein + zeaxanthin	267.0	338
Vitamin C	0.0	401
Thiamin	0.056	404
Riboflavin	0.414	405
Niacin	0.079	406
Vitamin B-6	0.118	415
Folate, total	30.0	417
Vitamin B-12	0.98	418
Choline, total	176.4	421
Vitamin K (phylloquinone)	3.8	430
Folic acid	0.0	431
Folate, food	30.0	432
Folate, DFE	30.0	435
Vitamin E, added	0.0	573
Vitamin B-12, added	0.0	578
Cholesterol	344.0	601
Fatty acids, total saturated	3.378	606

Fig. 5.4. Example Of Some Of The Nutrition Information Available In The T-FNDDS.

a Description Definition Languages (DDL) defining the syntax rules to combine, modify, and/or extend the descriptors. Most DDLs are based on the Extensible Markup Language (XML) (e.g. MPEG-7 [87], MPEG-21 [186], or Dublin Core [187]). By expressing the VCM in XML, the VCM can easily be exported to other applications. Figure 5.7 shows the VCM integrated into a web-based application.

As previously noted, in order to estimate energy and nutrients of foods and beverages from the food images, the FNDDS is used. However, the FNDDS does not provide all the information needed to determine the nutrients from the analysis of a food image. Once the volume estimate for a food item is obtained, it must be converted to a mass for calorie and nutrient estimation. In order to do this, the densities of the food items must be either known or estimated. For a number of foods, such



Visual Property Description	Visual Description	Value
Predominant Color 1 (RGB space, proportion, variance)	Pale Goldenrod	([238 232 170], 37%, 0.155)
Predominant Color 2 (RGB space, proportion, variance)	Gold	([255 215 0], 26%, 0.237)
Predominant Color 3 (RGB space, proportion, variance)	Honeydew	([240 255 255], 23%, 0.101)
Predominant Color 4 (RGB space, proportion, variance)	Bisque	([255 228 196], 11%, 0.065)
Type index	Solid	N/A
Shape Template	Irregular Polyhedric	0.712
Perceptual Feature 1 Description: Roughness	very low	0.109
Perceptual Feature 2 Description: Uniformity	medium	0.412

Fig. 5.5. Example VCM Integrated Into FNDDS To Provide Both Nutritional And Visual Description Of The Food Items. Using the XML Language Can Be Incorporated Into A Web-based Application Available To Researchers.

as plain yogurt, cake, apple and potato, a range of densities are reported in the literature, but there are many foods where this information is not available. Presently, there are more than 1000 "typical foods," such as a granola bar, with no density information in the FNDDS database. To address this problem, methods for accurately determining the density of foods have been developed using techniques such as computed tomography (CT), magnetic resonance imaging (MRI) and laser scanning. Stella *et. al.* in [188] and Kelkar *et. al.* in [189] presented several techniques for food density estimation.

We are also incorporating into the *T-FNDDS* Universal Product Code (UPC) information via food barcodes [190]. Having UPC information can provide faster and

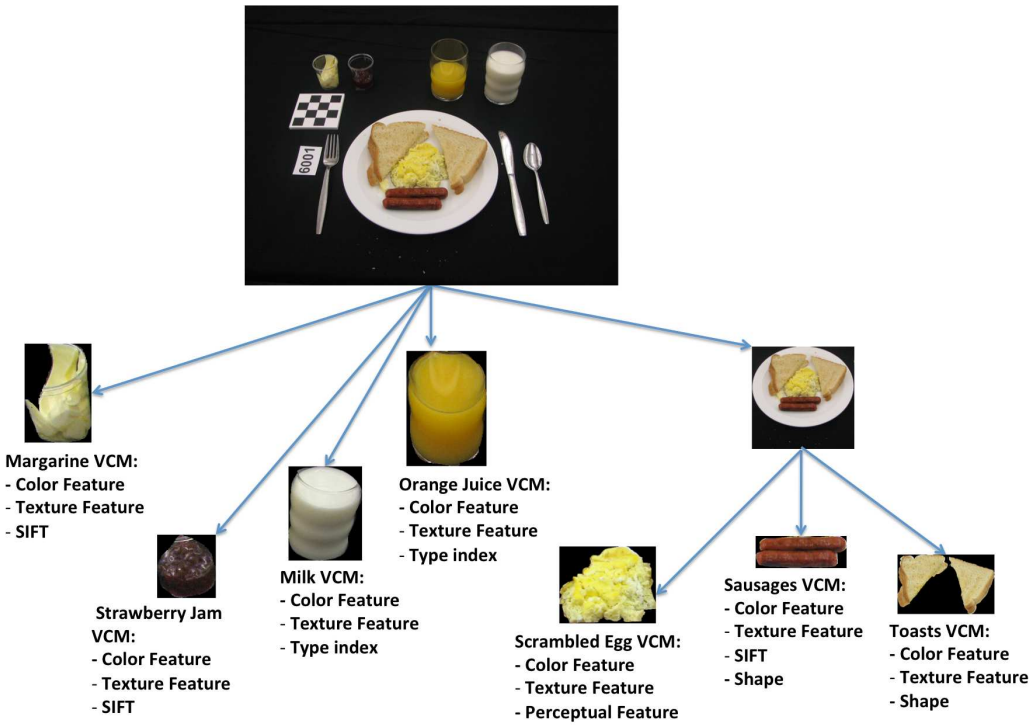
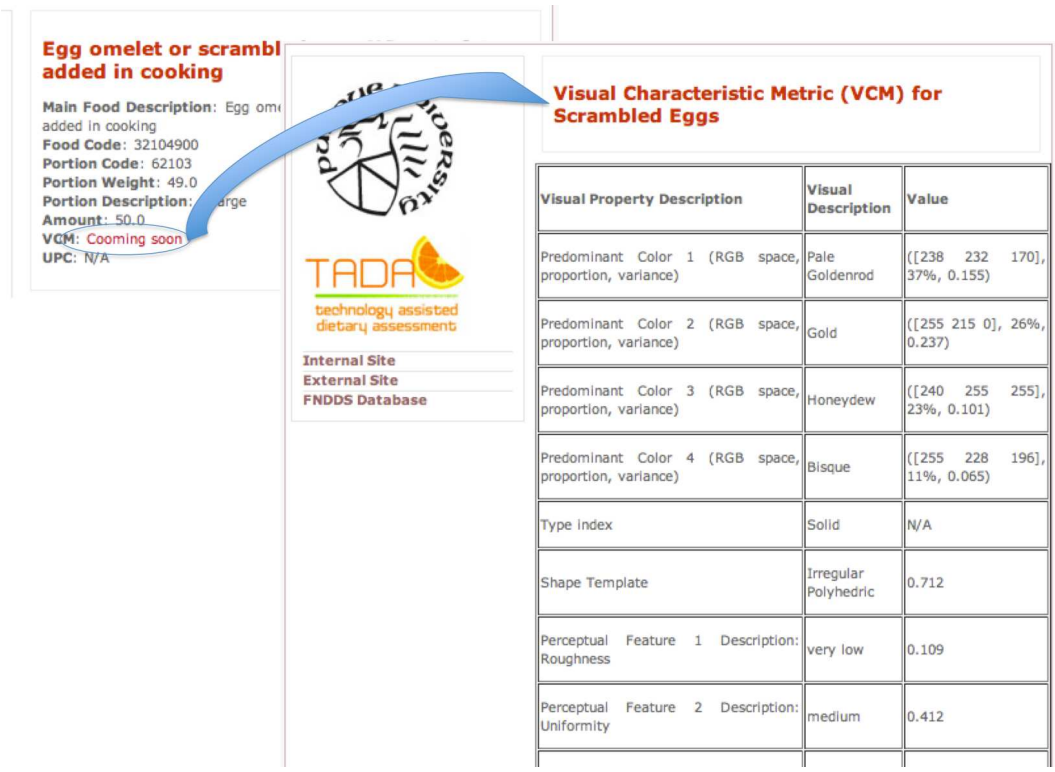


Fig. 5.6. Illustration of the Visual Characterization Metric (VCM) and the Structure for Visual Descriptions for a Food Image.

more accurate food identification in several situations when it is not possible for the user to take images of their food due to low-lighting conditions or when the user is in a hurry. Barcode information can provide precise food item identification and relevant portion size information (e.g., UPC number 04963406 provides the identification of a 12 oz can of Coca-Cola Classic). The barcode can also be used to correct incorrect identification of foods in the review process (step 4 of Figure 1.2). Finally, by integrating the UPC information into the FNDDS, the healthcare professional has access to other types of information such as the food ingredients or the composition of the food not available in the original FNDDS database. Figure 5.8 illustrates the barcode integration in the *T-FNDDS*.



Visual Property Description	Visual Description	Value
Predominant Color 1 (RGB space, proportion, variance)	Pale Goldenrod	([238 232 170], 37%, 0.155)
Predominant Color 2 (RGB space, proportion, variance)	Gold	([255 215 0], 26%, 0.237)
Predominant Color 3 (RGB space, proportion, variance)	Honeydew	([240 255 255], 23%, 0.101)
Predominant Color 4 (RGB space, proportion, variance)	Bisque	([255 228 196], 11%, 0.065)
Type index	Solid	N/A
Shape Template	Irregular Polyhedric	0.712
Perceptual Feature 1 Description: Roughness	very low	0.109
Perceptual Feature 2 Description: Uniformity	medium	0.412

Fig. 5.7. An Example of a VCM Integrated into FNDDS to Provide Both Nutritional and Visual Description of the Food Items. Using XML the VCM Can Be Incorporated into a Web-based Application.

5.1.3 E-TADA

The purpose of *E-TADA* or the *Experiments database* is to store data from the nutritional studies we have conducted that is related to a user (see Footnote 1 in Chapter 5). To date we have conducted two types of user studies: free-living studies and controlled studies. During free-living studies users carry on their normal lives and acquire food images as they would in normal life scenarios such as eating at fast-food dinners, eating at home watching TV, eating at the office, eating on-the-go, and other real life situations. In controlled studies participants eat their meals in a controlled environment including specific foods provided for each meal.

We store the following personal information in *E-TADA*:

- Participant ID

Fig. 5.8. Barcode Information Integrated Into The T-FNDDS.

- Date of birth
- Visit date
- Age
- Gender
- Height
- Weight
- Ethnicity
- Race

PARTICIPANT INFORMATION
Participant ID: 8037
Visit Date: 2011-07-05 (26 years 5 months and 3 days)
Date of Birth: 1985-02-02
Age: 26 years 5 months and 3 days
Gender: male
Height: 1.81 m.
Weight: 98.1 Kg.
Ethnicity: Hispanic
Race: white
Pregnant: no
Zipcode: 47901

STUDIES PARTICIPATED:

Further Study Information(1): Free Living 2011 Spring

All Study (1) Images: Images from Free Living 2011 Spring

Data Files: Participant 8037

- Food Description Components File
- Single Food File
- Eating Event File
- Total Day File

PARTICIPANT/STUDY INFORMATION:
Participant ID: 8037
Study: Free Living 2011 Spring
Study images: Images_Free Living 2011 Spring

STUDY FACTS FOR THIS PARTICIPANT

Most Frequent Foods Eaten:
water
milk
salad mix
lasagna
english muffin
garlic toast
ham sandwich
potato chips
ranch dressing
beer

Meal Averaged Time:12.95 minutes

Meal Occasions:
2011-07-11_12-10-59: 10000
2011-07-11_20-21-34: 10018
2011-07-12_00-22-52: 10026
2011-07-12_12-10-38: 10042
2011-07-12_07-29-30: 10054
2011-07-12_17-59-20: 10058
2011-07-12_22-16-36: 10072

Fig. 5.9. Examples of Data Available for Each User in *E-TADA* for the Free-living Studies.

- Zip code

Some other information is study specific, such as menus, dates and times, or number of users.

As it is discussed in section 5.2.2, *E-TADA* can be an excellent source of information to predict eating patterns with respect to a user or entire populations [185]. User specific statistics can be estimated from the data stored in *E-TADA* including the most frequently eaten foods by a user, average eating occasion duration time, most frequent eating locations, and time when a meal was consumed, among other. Figures 5.9 and 5.10 show examples of some of the data available for free-living and controlled studies respectively.

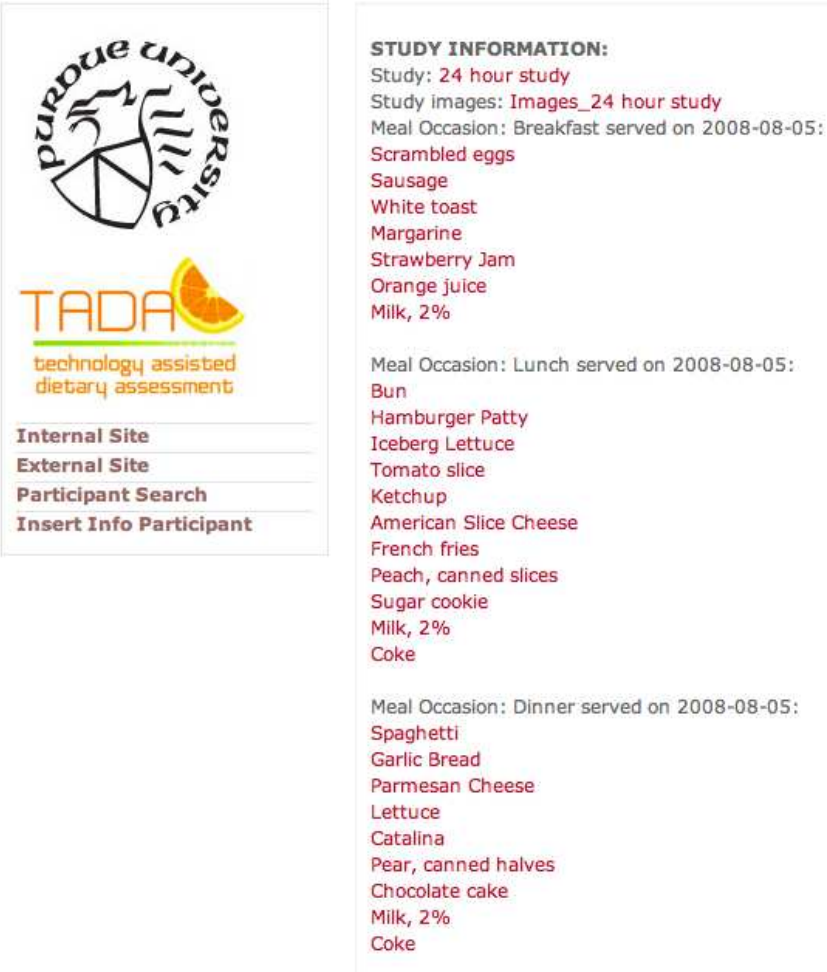


Fig. 5.10. Examples of Data Available for a Controlled Studies Including Menus Served.

5.2 TADA System Architecture

5.2.1 Mobile User Interface

In this section we describe our mobile user interface and its main features and modes of operation of our system as of late 2011. The implementation of the user interface has been a collaborative work with my colleagues SungYe Kim and Ziad Ahmad. As shown in Figure 1.2, the TADA system utilizes a client-server configuration. Our strategy takes into consideration four elements in the user interface: easy and

fast collection of daily food images, minimum user interaction, protection of personal data, and network status.

There are three modes of operation of the mdFR: Record Eating Events mode, Review mode, and Alternate Method mode.

Record Eating Events Mode

This mode allows users to quickly capture images of their eating occasions. Figure 5.11.a shows the main view of this mode, with two main buttons, one for taking the "before eating" image and another for taking the "after eating" image. In order to guarantee that the images acquired can be analyzed at the server, a fast quality check is done after the acquisition of each image. A series of tips and reminders are available to provide users with guidelines to acquire images. After the acquisition of the "after eating" image the mobile device checks the network status, if there is Wi-Fi/3G/4G coverage the images are sent to the server along with metadata. In the case where there is no coverage, a button is displayed on the right top of the screen to indicate the number of unsent eating occasions. At a later time, as soon as there is network coverage, the images can be sent to the server.

Review Mode

After recording eating occasions and the images being analyzed at the server, the users can review the analyzed results from the server to either confirm or change food types (food names). The users request the results from the server and if available these are sent to the mobile device. A list view is populated on the mobile device with the unconfirmed results showing a thumbnail of the before eating image and the date and time of the eating occasion (Figure 5.11.b). The current reviewing process consists of reviewing the food types identified by the server. The image is displayed on the device's display where foods are labeled with a bubble and pin system (Figure 5.11.c). Pins can be removed, added, or edited accordingly. Once items have been confirmed

by the user, he/she sends the updated results back to the server. At the server a further image analysis refinement and volume estimation is performed according to the user's feedback.

Alternate Method Mode

The Alternate Method manages eating events when food images are not acquired. There are situations where it may be impossible for a user to capture food images, *e.g.*, low lighting conditions, driving, on-the-go situations. Previously, a prototype of the Alternate Method in [191] was described, which consisted of providing the users with tools for creating eating occasions containing enough information for nutrient analysis, including date and time, food name, measure description and the amount of intake. We are currently working on a new design of the Alternate Method in order to take full advantage of the new features offered by current mobile telephones, including actions such as using barcodes, taking quick typed notes on the mobile device, and even recording voice notes.

5.2.2 TADA System Integration

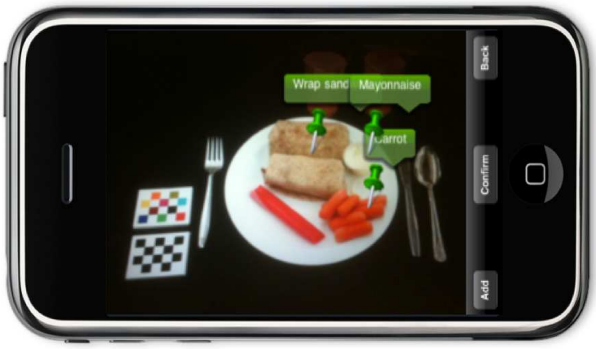
All the operational characteristics of the TADA system are coordinated by the *I-TADA, T-FNDDS*, and *E-TADA* databases. This includes the information exchanged between the client (the mobile telephone) and the backend server system, the image analysis and communications layer at the server, and finally, the web-based interface to the databases. Figure 5.12 illustrates the high-level interactions of the databases with the rest of the TADA system.

Communications and Image Analysis Integration

Data between user's mobile device and server is transferred via Wi-Fi or cellular network, *e.g.* 3G/4G network. Typically, our message structure is as follows:

(a) Record Eating Events Mode - Home Screen

(b) Review Mode - List View



(c) Review Mode - Review Results

Fig. 5.11. Examples Of The Mobile User Interface For Dietary Assessment (mdFR). (a) Home Screen - Record Eating Events Mode, (b) List View Of All Unconfirmed Eating Occasions - Review Mode (c) Review Results- Review Mode.

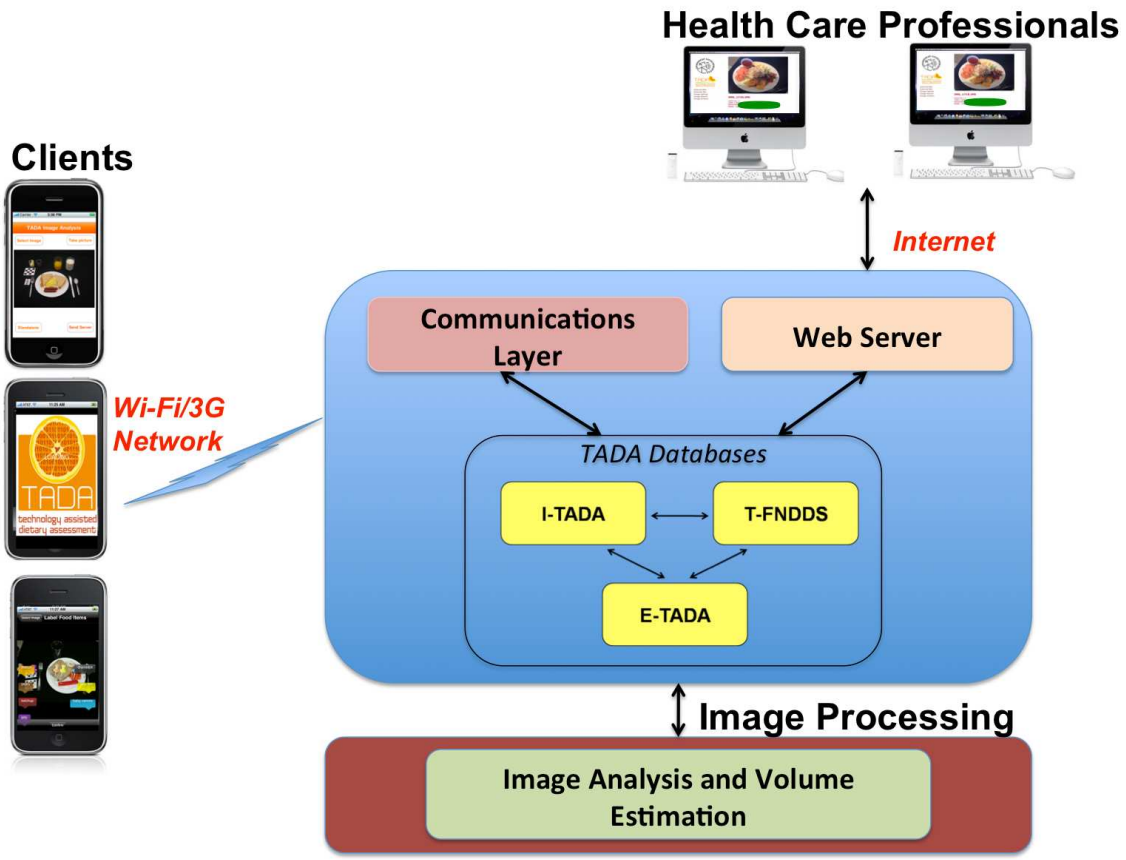


Fig. 5.12. Databases interaction with the TADA system.

- Content Type: specifies the content of a message.
- Participant ID: this field contains the participant identification code.
- Action: to let the receiver know what to send next.
- Device UDID: the device identification code.
- Encrypted Password: an encrypted password.
- Body: message content.

There are four types of communications between the client and the backend server. First, the mobile device sends the "before" and "after" eating occasion images and metadata to the server. Second, results of the image analysis are sent from the server

to the mobile device for user revision and confirmation. Third, these confirmed results are sent back to the server. Finally, reminder messages are sent from the server to the mobile device for various purposes. Except for the reminder messages, the other three communication types are initiated by the mobile device with the server actively "listening." The security and privacy of the data generated by the mdFR are critical. For this reason, all users in our studies are uniquely identified with a code - names, address, and other identifying information is not stored in any of our databases.

The image processing and analysis is done in a separate server connected to the main database server. This distributed architecture guarantees a more cost-efficient approach and an easier path for expansion and scalability. The *I-TADA* coordinates the data flow between the communications layer of the main server with the image analysis server. As soon as a new pair of images are received, they are sent to the image processing server for analysis.

We have designed an architecture that is cross-platform compatible. This allows different types of mobile devices to share the same server side resources. Data between the mobile device and the server is transferred via Wi-Fi or cellular 3G network. Our communications protocol is a message oriented network protocol that runs over a bidirectional stream (TCP socket) allowing peers (clients and the server) on either end to send messages in both directions. Messages are multiplexed over the connection so several may be in transit at once. This guarantees that a long message (e.g. a food image) will not block the delivery of others.

Web Interface Application

The use of a web-based interface to the TADA databases allows healthcare professionals to gain access to the data in real time anywhere. This provides access to a wide range of information including, nutritional food information, user statistics, and image analysis evaluation.

Typically there are three main elements in any web application architecture: a relational database, middleware, and web server (Appendix A). The relational databases (*I-TADA*. *T-FNDDS*, and *E-TADA*) described above have been developed using the PostgreSQL language, based on a SQL syntax [192]. The middleware consists of a class of programming modules that work closely with the web server to interpret the requests made from the web interface application, interact with other programs on the server to fulfill the requests, and then, indicate to the web server exactly what resources to serve to the web interface application in the internet browser. Finally, the web server listens for requests from the web interface and response to these requests after executing the tasks.

One important purpose of the web application is to provide healthcare professionals with a mechanism to monitor users in nutritional studies. From any internet browser there is access to the user's data from past and current studies. As soon as the user acquires the images of an eating occasion and sends them to the server, these are available to the healthcare professional to examine and/or monitor the food labeling adjustments done by the user. Real-time statistics can also be obtained, such as the most common eaten foods by a user, average eating occasion duration time, and time when an eating occasion was consumed. Healthcare professionals can also access the menus used in the case of controlled studies, and nutritional information of each of the foods in the image that the user recently acquired. Figure 5.13 shows an example of some of the information available from the web application for a user. If the healthcare professional is interested in seeing the location where an image was acquired we have connected the geolocation information using Google Maps (Figure 5.14).

All the data is connected so that information can be accessed from all three databases. From a web browser, one can select any image in the database and display the information related to the image. Each set of information has a hyperlink associated that executes a particular query in the database and displays the results. For example, if one clicks on the date that the image was acquired, a list of images

All participants with gender: male

8053
8048
8047
8044
8040
8037
8032
8028
8027
8019
8005
8004
6006
6003
6002
13
1001
0991
0927
0925
0915
0914
0913
0911
0910
0908
0903

Most Frequent Foods Eaten:

water
milk
salad mix
lasagna
english muffin
garlic toast
ham sandwich
potato chips
ranch dressing
beer

Meal Averaged Time:12.95 minutes

All participants on 2008-08-05:

6006
6003
6002
6001

Fig. 5.13. Examples of User Specific Statistics Available via the Web Interface.

taken on that same day will be displayed. Similarly, one can select a user and examine the user's information and use the links to view other users with similar characteristics, e.g. age, gender (Figure 5.13), weight and geographical area. Metadata related to the image is also shown: EXIF data and geolocation coordinates are displayed with embedded hyperlinks (e.g. all the images taken with the same camera or the geographic area where the image was acquired).

Complementing the web interface, we have created an email-based monitoring system such that when a particular event occurs, an email is automatically generated and sent to the healthcare professional with the event notification information and instant access to all the user's information. These events may vary depending on the characteristics of the study. Examples of events that can be notified via email are

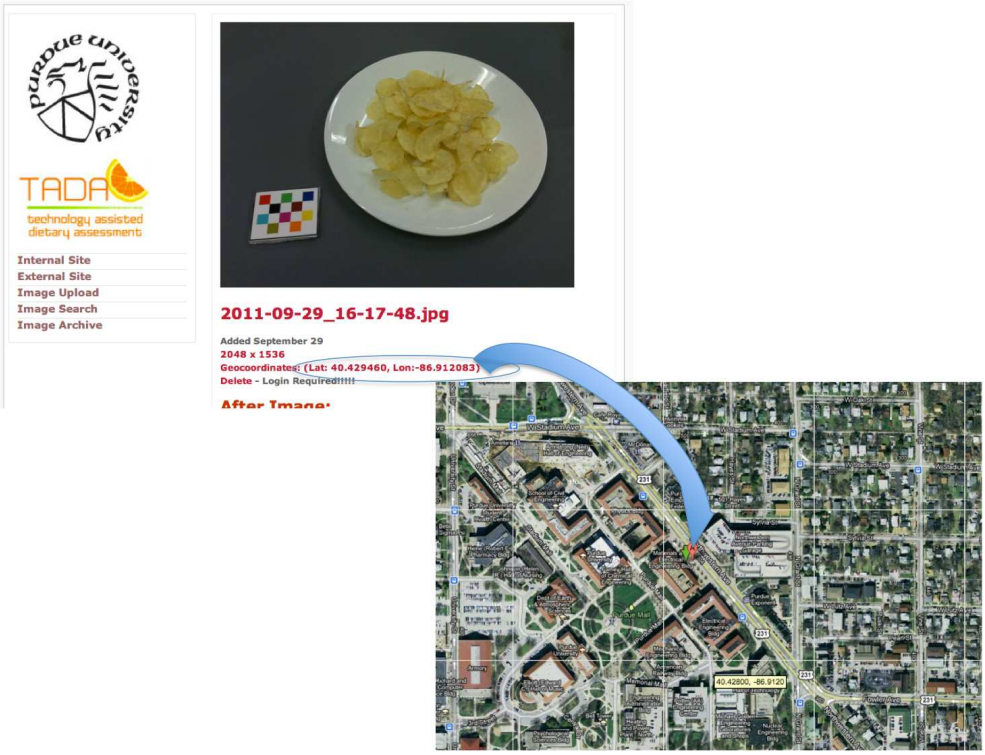


Fig. 5.14. GPS Location Connected To An Online Map Service For Visualization.

user inactivity or when a user consumes certain types of foods. Figure 5.15 shows an example of an email notification.

5.2.3 The TADA Databases: Current Deployment

Our current database system as of late 2011 consists of 119 tables with more than 600 fields and 15 million parameters. Overall, there are more than 11,500 food images in *I-TADA*. In the last 15 months more than 6,400 images have been sent to the system using the mdFR. In the *I-TADA* there also are 7,400 groundtruth images. More than 170 participants have used the mdFR in the past three years. There are 7,000 approximately food items in our database.

Appendix A lists all software and modules required for our database deployment.

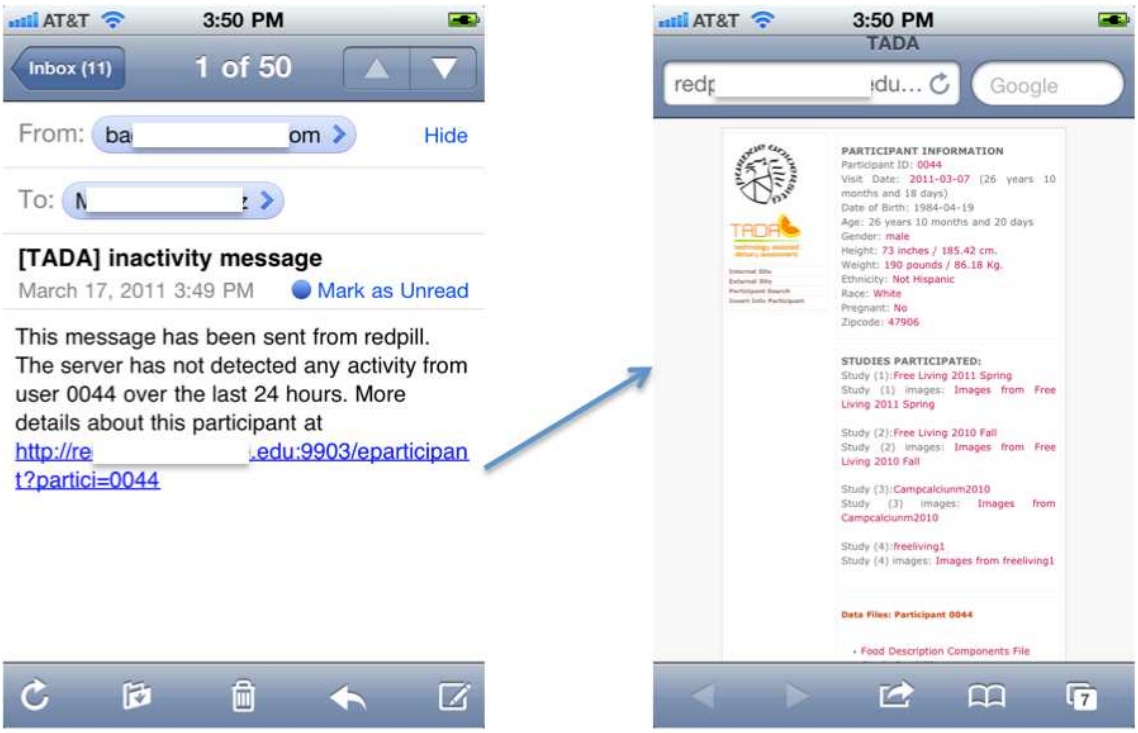


Fig. 5.15. Example Of Email Notifications Sent To The Researcher With Participant's Information.

5.2.4 User Study Validation

Several nutritional studies were conducted by the Department of Foods and Nutrition at Purdue University whereby participants were asked to acquire images with the mdFR mobile application, and review the results from the automatic analysis. We have tested our system with adolescents and adults of different ages and background characteristics under controlled and free living conditions. More than 6000 images have been acquired by participants using the proposed TADA system.

These studies are particularly useful in order to evaluate the users' attitudes towards the mdFR [193–195]. Their feedback is important to design the next generation of user interface on the mobile device.

In Section 4.5 we presented results for food identification from our user study conducted during the Spring and Summer of 2011. In this section, we evaluate the

study from the point of view of the user and his/her interaction with the mdFR through the mobile application. We were interested in the interaction between the user and the review process after receiving the food classification results from the server.

In the review process, users, through a series of intuitive pin manipulation actions, review the food identification results by confirming a pin (the food was correctly identified), removing a pin (the pin was pointing to a non-food item), adding a pin (the food is not in our database, or it was not detected by our segmentation and classification system), or changing a pin. In this last option, there are two cases that we were particularly interested in that it can help evaluate the automatic food identification process. (1) The user changes the pin and selects a food from the list of suggested items from the classifier output, or (2) the user changes the pin and selects a food from the general list of foods. The list of suggested items consists of a list of 4 food names that the automatic image analysis considered as the potential top 4 food classes corresponding to each pin. Table 5.1 summarizes the pin manipulation actions of the top 4 users in terms of correct classification accuracy as it is shown in Figure 4.16, bottom 4 users, and the average of all users. Although these results do not necessarily need to reflect the performance of the automatic identification process, they do correlate with it. Pin manipulation can translate into user satisfaction towards our application. Some users may opt for removing all pins when there are many pins on the screen due to different food segmentations even though most of the pins are correctly labeled. Other users may not use the suggested food items list and they go straight towards the general list. Overall, these pin interactions are more valuable information to evaluate the user interface of the review mode and how users interact with it than a realistic tool to measure food identification performance.

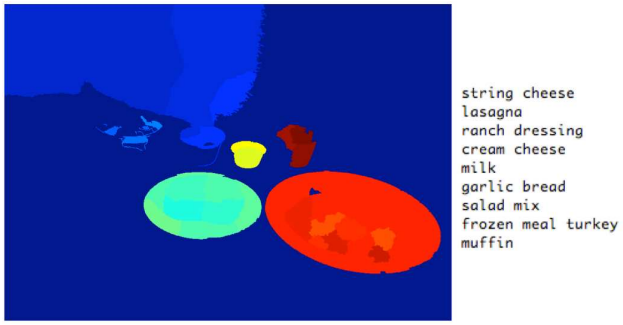
Once the results have been reviewed and/or modified, they are sent back to the server. As a result of removing pins some segmented regions become unlabeled, therefore segmentation masks are incomplete. Complete masks for each food item are necessary for accurate image analysis. One way to resolve this problem is to

Table 5.1
Pin manipulation actions of the top 4 users, bottom 4 users and average of all users in the Spring/Summer 2011 free living study. **Added** refers to pins added by the user due to mis detection of food items in the segmentation step, or foods not in our database, **Removed** pins removed by the user because they are pointing to a non food item, **Confirmed** refers to all pins that were correct, **Suggested** pins that were changed but the correct food was within the suggested foods, and finally, **Changed** refers to pins that were changed and not in the suggested foods list.

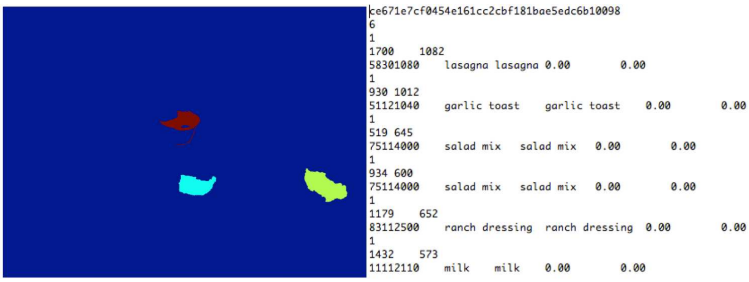
Participant	**Added Pins**	**Removed Pins**	**Confirmed Pins**	**Suggested Pins**	**Changed Pins**
Average	**2.28**	**3.90**	**1.07**	**1.95**	**0.68**
Top 1(8061)	2.51	0.61	3.25	5.9	2.06
Top 2 (8028)	0.84	4.92	1.05	1.92	0.67
Top 3 (8025)	1.09	3.95	0.67	1.22	0.43
Top 4 (8051)	1.36	0.75	4.02	7.31	2.54
Bottom 1 (8003)	1.23	4.13	0.20	0.40	0.14
Bottom 2 (8048)	1.03	3.96	0.64	1.15	0.40
Bottom 3 (8021)	2.78	3.67	0.99	1.81	0.63
Bottom 4 (8023)	3.75	3.62	0.20	0.38	0.12

classify again the original segments using the user's feedback, *i.e.* the true food labels. The classifier can be constrained by only considering the labels confirmed by the user. The classification task becomes easier since, usually, only 5 or 6 classes can appear in the same image. We can improve the initial classification step and label all the segmented regions that the user removed, therefore obtaining the complete masks. Figure 5.16.a shows an example of a mask after food identification (data sent to user). Figure 5.16.b shows an example of a mask resulting from the user's feedback where some foods have been added and other pins removed. Figure 5.16.c shows the final mask resulting from image analysis refinement by using the user's feedback, the suggested foods list, and nearest neighbor classifier.

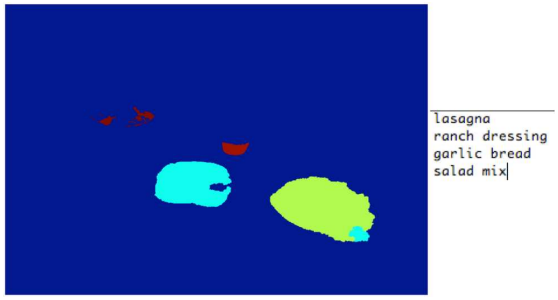
(a) Original Image



(b) Mask Resulting From Segmentation With Labels



(c) User Feedback (Mask And Food Labels)



(d) Mask After Classification Refinement With Labels

Fig. 5.16. An Example Of Classification Refinement Using User's Feedback. (a) Original Image, (b) Mask After Image Analysis, (c) User Feedback (Mask And Local Labels), And (d) Mask After Classification Refinement.

6. CONCLUSIONS AND FUTURE WORK

In this thesis an imaging based tool to assess diet is described. In particular, methods for automatic food identification and the design and deployment of an integrated food image-based database system have been investigated.

6.1 Conclusions

We have focused our efforts on designing features for object detection in order to visually characterize objects in a scene. The main contributions in the area of food identification are as follows:

- We investigated several color features from various color spaces. Local and global color features have been proposed in order to characterize many types of objects such as objects with homogeneous color distribution and complex objects composed by primary components (ingredients) with different color distributions. The color features have shown to be robust to color perturbations that increase intra-class variance (e.g. green vs. ripe fruit).

- We proposed three unique texture descriptors namely the Entropy-based categorization and Fractal Dimension estimate (EFD), the Gabor-based image decomposition and Fractal Dimension estimate (GFD), and the Gradient Orientation Spatial-Dependence Matrix based on the spatial relationship of gradient orientations (GOSDM).

- We examined local descriptors in order to form visual vocabularies.

In the classification process we have proposed a multichannel classification system. The main contributions are as follows:

- We investigated late decision fusion in order to combine multiple feature channels as opposed to fusing feature spaces into a higher dimensional feature space.
- We derived confidence scores for SVM and KNN classifiers in order to combine multiple feature spaces into one class decision. These proposed scores have also been used to improve the image segmentation.
- We explored the use of contextual information to refine the classifier decision by developing a model to obtain a labeling agreement based on object interaction and misclassification rates.

Each of the fundamental components that constituted our classification model: codebook refinement of the local features, late decision fusion, and contextual information made contributions to the final classification results. Late decision fusion can be seen as a major factor in the final classification performance for large number of class candidates and very distinct feature spaces (e.g. color, texture, local features). The proposed contextual information has also proven to be an efficient approach to correct misclassifications based on the classifier's final decision. Finally, the codebook refinement of the local features has proven less successful. We found that for visually inhomogeneous foods, projecting the local features into lower dimensional feature spaces was not as efficient as considering the original visual codebook.

We also proposed an integrated food image-based database system for our mobile device food record system. We extended our database by developing a web-based application that allows researchers to monitor patients in real-time. We enhanced traditional nutritional databases with data obtained from food images as follows:

- We proposed an integrated food image-based database system where data from users and images is connected to nutritional information for dietary assessment.
- We created a web-based interface for nutritionists, dietitians, and researchers for guidance and monitoring of patients.
- We extended the USDA FNDDS to include image related data.

- We created a cooperative platform to explore and discover embedded patterns in dietary habits.

6.2 Future Work

One focus of this work has been to find texture descriptors that can describe the texture patterns found in foods. In this thesis we describe three different texture descriptors (eg. EFD, GFD, GOSDM) that we have proposed in [71]. Although each outperforms widely used texture descriptors such as Gabor features, and GLCMs, they do not achieve scale invariance. The first step on developing scale invariant food texture descriptors would be to measure the scale ranges that are typically observed in images acquired by users. Using our collection of more than 11000 images we can estimate the size (number of pixels) that a common known object to all the images (e.g. fiducial marker) measures on average, and thus obtain the ratio between physical dimensions and number of pixels for the other objects in the images. Besides, the average, standard deviation values and maximum and minimum values would be required to formulate a food scale model. These values can be used to develop scale invariant versions of our texture descriptors.

Unfortunately, state-of-the-art object classification approaches cannot distinguish between 6000 or 7000 types of foods. One way to extended our work considering hundreds of food classes is to constrain the problem by exploring more types of contextual information and optimizing the training dataset (individual/personalized classifiers). Up to this point we have considered object combination likelihoods, misclassification rate information, and the classifier confidence as potential contextual information. In our particular application, there is other contextual information that we are currently exploring such as image metadata, GPS information, time/date of the eating occasion, and dietary habits that could be used for a more efficient food/object classification (Figure 6.1).

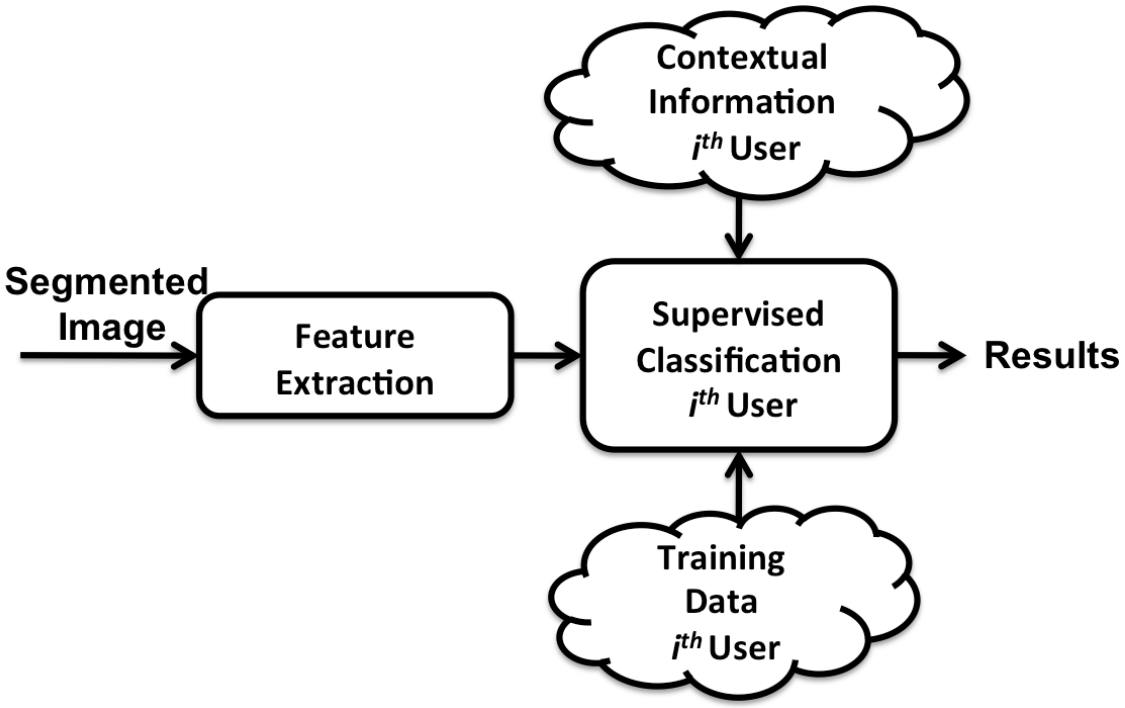


Fig. 6.1. Block Diagram Of A Personalized Classifier Using Contextual Information.

In our food classification scheme there is a scaling problem that needs to be addressed. There are many foods available and current classification approaches may be infeasible for a large number of training classes. One way to address large food datasets is to consider "temporal" contextual information (eating patterns). By having an individualized/personalized training dataset consisting of limited initial training data (e.g., the most common foods eaten) and dynamically adding foods and beverages commonly eaten by the particular individual.

Selection of optimal training data is still an open problem. Overall, we believe that selecting a large amount of samples with large visual variance for training is a key factor in achieving high classification rates under real-life environments. Feature selection, sample selection, or cross domain learning methods provide an initial step towards optimal training data. Using previously gathered data from the user as training data will increase the success of our methods.

(a) Top Retrieved Images For The Query *Brownie* From Google Search

(b) Top Retrieved Images For The Query *Lasagna* From Google Search

Fig. 6.2. Examples Of Training Images Retrieved With Google Search (a) *Brownie*, and (b) *Lasagna*.

Another aspect that should be explored is the source of the training data. Millions of images are available on internet. By using text queries with online search engines we can retrieve large amounts of ranked images that can constitute/complement our training dataset. By using a sample selection process on these data, more robust and large training data models can be built. As an example Figure 6.2 shows the top retrieved images from the queries *Brownie* and *Lasagna* respectively.

Another issue is the case of complex foods made with many ingredients that require training preselected subcategories manually. For example, a *cheeseburger* is composed of a *bun, cheese, burger patty, lettuce, tomato,....* From the segmentation these different components will be part of different segmented regions. Should all these individual segmented regions be included in the training data corresponding to the cheeseburger, or only the entire burger segmented region should be part of our training set? Should we design a specific grammar mechanism that relates all these ingredients together to form the cheeseburger in a part-based classification approach? These are some of the questions currently being investigated. We recommend the following should be explored further:

- Explore other models of contextual information that are particular to each user by means of geolocation, date, time data from images, and eating patterns.
- Optimize the training data by extending the current classification model considering personalized classifiers (including foods consumed by each individual).
- Investigate the formulation of pattern grammar syntax to identify complex food items by capturing hierarchical, and compositional patterns, to include some probabilistic reasoning and inference mechanisms for labeling decision.
- Investigate multiple kernel learning approaches for feature space combination and compare it with decision fusion.

6.3 Publications Resulting From This Work

Journal Articles:

1. **Marc Bosch**, Nitin Khanna, Carol J. Boushey, and Edward J. Delp, "An Integrated Image-Based Food Database System with Application in Dietary Assessment," *IEEE Transactions on Information Technology in Biomedicine*, submitted.

2. Fengqing Zhu, **Marc Bosch**, Nitin Khanna, Carol J. Boushey and Edward J. Delp, "Multiple Hypothesis Image Segmentation and Classification with Application to Dietary Assessment," *IEEE Transactions on Image Processing*, submitted.

3. Fengqing Zhu, **Marc Bosch**, Insoo Woo, SungYe Kim, Carol J. Boushey, David S. Ebert, Edward J. Delp, "The Use of Mobile Devices in Aiding Dietary Assessment and Evaluation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 4, pp. 756-766, August 2010.

4. Bethany L. Six, TusaRebecca E. Schap, Fengqing Zhu, Anand Mariappan, **Marc Bosch**, Edward J. Delp, David S. Ebert, Deborah A. Kerr, Carol J. Boushey, "Evidence-Based Development of a Mobile Telephone Food Record," Journal of American Dietetic Association, January 2010, pp. 74-79.

Conference Papers

1. Fengqing Zhu, **Marc Bosch**, Ziad Ahmad, Nitin Khanna, Carol J. Boushey, Edward J. Delp, "Challenges in Using a Mobile Device Food Record Among Adults in Freeliving Situations," *mHealth Summit*, December, 2011, Washington D.C.

2. **Marc Bosch**, Fengqing Zhu, Nitin Khanna, Carol J. Boushey, Edward J. Delp, "Combining Global and Local Features for Food Identification and Dietary Assessment," *Proceedings of the IEEE International Conference on Image Processing*, Brussels, Belgium, September 2011.

3. Fengqing Zhu, **Marc Bosch**, N. Khanna, Carol J. Boushey, Edward J. Delp, "Multilevel Segmentation for Food Classification in Dietary Assessment," *Proceedings of the International Symposium on Image and Signal Processing and Analysis*, Dubrovnik, Croatia, September 2011.

4. **Marc Bosch**, Fengqing Zhu, Nitin Khanna, Carol J. Boushey, Edward J. Delp, "Food Texture Descriptors Based on Fractal and Local Gradient Information,"

Proceedings of the European Signal Processing Conference (Eusipco), Barcelona, Spain, August 2011.

5. **Marc Bosch**, TusaRebecca E. Schap, Nitin Khanna, Fengqing Zhu, Carol J. Boushey, Edward J. Delp, "Integrated Databases for Mobile Dietary Assessment and Analysis," *Proceedings of the 1st IEEE International Workshop on Multimedia Services and Technologies for E-Health in conjunction with the International Conference on Multimedia and Expo (ICME)*, Barcelona, Spain, July 2011.

6. Fengqing Zhu, **Marc Bosch**, Nitin Khanna, TusaRebecca E. Schap, Carol J. Boushey, David S. Ebert, Edward J. Delp, "Segmentation Assisted Food Classification for Dietary Assessment", *Proceedings of Computational Imaging IX, IS&T/SPIE Electronic Imaging*, San Francisco, CA, January 2011.

7. SungYe Kim, TusaRebecca E. Schap, Marc Bosch, Ross Maciejewski, Edward J. Delp, David S. Ebert, and Carol J. Boushey, "A Mobile User Interface for Image-based Dietary Assessment", *Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia*, Limassol, Cyprus, December 2010.

8. **Marc Bosch**, Fengqing Zhu, TusaRebecca E. Schap, Carol J. Boushey, Deborah Kerr, Nitin Khanna, Edward J. Delp, "An Integrated Image- Based Food Database System with Application in Dietary Assessment," *mHealth Summit*, November, 2010, Washington D.C. *(Meritorious New Investigator Award)*

9. Fengqing Zhu, **Marc Bosch**, Carol J. Boushey, and Edward J. Delp, "An Image Analysis System for Dietary Assessment and Evaluation," *Proceedings of the IEEE International Conference on Image Processing*, September, 20010, Hong Kong.

10. Anand Mariappan, **Marc Bosch**, Fengqing Zhu, Carol J. Boushey, Deborah A. Kerr, David S. Ebert, Edward J. Delp, "Personal Dietary Assessment Using

Mobile Devices," *Proceedings of the IS&T/SPIE Conference on Computational Imaging VII*, Vol. 7246, San Jose, January 2009.

LIST OF REFERENCES

LIST OF REFERENCES

[1] National Vital Statistics System United States 2006 and 2007, "Quickstats: Age-adjusted death rates for the 10 leading causes of death," *Morbidity and Mortality Weekly Report*, vol. 58, no. 46, p. 1303, November 2009.

[2] United States Department of Agriculture, "Report of the Dietary Guidelines Advisory Committee on the Dietary Guidelies for Americans," Available: http://www.cnpp.usda.gov/DGAs2010-DGACReport.htm, 2001.

[3] F. Thompson, A. Subar, C. Loria, J. Reedy, and T. Baranowski, "Need for technological innovation in dietary assessment," *Journal of the American Dietetic Association*, vol. 110, no. 1, pp. 48–51, 2010.

[4] C. Ogden, M. Carroll, L. Curtin, M. McDowell, C. Tabak, and K. Flegal, "Prevalence of overweight and obesity in the united states, 1999-2004," *Journal of the Medical American Association*, vol. 295, no. 13, pp. 1549–1555, April 2006.

[5] P. Muenning, E. Lubetkin, H. Jia, and P. Franks, "Gender and the burden of disease attributable to obesity," *Journal of Public Health*, vol. 96, no. 9, pp. 1662–1668, September 2006.

[6] C. Ogden, K. Flegal, M. Carrol, and C. Johnson, "Prevalence and trends in overweight among US children and adolescents, 1999-2000,," *Journal of the Medical American Association*, vol. 288, no. 9, pp. 1728–1732, October 2002.

[7] A. Fagot-Campagna, J. Saadinem, K. Flegal, and G. Beckles, "Diabetes, impaired fasting glucose, and elevated hbalc in us adolescents: the third national helath and nutrition examination survey," *Diabetes Care*, vol. 24, pp. 834–837, 2001.

[8] M. Uusitupa, "Early lifestyle intervention in patients with non-insulin-dependent diabetes mellitus and impaired glucose tolerance," *Ann. Med.*, vol. 28, no. 5, pp. 445–449, October 1996.

[9] L. Bandini, A. Must, H. Cyr, S. Anderson, J. Spaldano, and W. Dietz, "Longitudinal changes in the accuracy of reported energy intake in girls 10-15 years of age," *The American Journal of Clinical Nutrition*, vol. 78, pp. 480–484, 2003.

[10] A. Prentice, A. Black, W. Coward, and et al., "High levels of energy expenditure in obese women," *British Medical Journal*, vol. 292, pp. 983–987, 1986.

[11] C. J. Boushey, D. Kerr, J. Wright, K. Lutes, D. S. Ebert, and E. J. Delp, "Use of technology in children's dietary assessment," *European Journal of Clinical Nutrition*, vol. 63, pp. 550–557, 2009.

[12] J. Obermayer, W. Riley, O. Asif, and J.Jean-Mary, "College smoking-cessation using cell phone text messaging," *Journal of American College Health*, vol. 53, no. 2, pp. 71–78, 2004.

[13] A. Rodgers, T. Corbett, D. Bramley, D. Riddell, M. Wills, R. Lin, and et al, "Do u smoke after txt? results of a randomised trial of smoking cessation using mobile phone text messaging," *Journal of Tobacco Control*, vol. 14, no. 4, pp. 255–261, 2005.

[14] V. Franklin, A. Waller, C. Pagliari, and S. Greene, "A randomized controlled trial of sweet talk, a text-messaging system to support young people with diabetes," *Journal of Diabetic Medicine*, vol. 23, no. 12, pp. 1332–1338, 2006.

[15] R. Hurling, M. Catt, M. D. Boni, B. Fairley, T. Hurst, P. Murray, and et al., "Using internet and mobile phone technology to deliver an automated physical program: randomized controlled trial," *J. Med Internet Res.*, vol. 9, no. 2, p. e7, 2007.

[16] "Calorie Counter and Diet Tracker." [Online]. Available: http://www.myfitnesspal.com/welcome

[17] "Lose It!" [Online]. Available: http://www.loseit.com

[18] "Calorie Smart." [Online]. Available: http://www.coheso.com/caloriesmart-d.html

[19] "Restaurant Calorie Counter." [Online]. Available: http://www.euliax.com

[20] "DrinkFit." [Online]. Available: http://www.drinkfitapp.com

[21] "Fast Food Calorie Checker." [Online]. Available: http://www.euliax.com

[22] "Calorie Traker." [Online]. Available: http://www.livestrong.com/thedailyplate/iphone-calorie-tracker/

[23] "DailyBurn." [Online]. Available: http://www.dailyburn.com

[24] "Food IQ." [Online]. Available: http://www.myfoodiq.com

[25] "Tap&Track Calorie, Weight and Exercise Tracker." [Online]. Available: http://nanobitsoftware.com

[26] "iPhone Calorie Counter," 2010, Demand Media, Inc. [Online]. Available: http://www.livestrong.com

[27] P. Jarvinen, T. Jarvinen, L. Lahteenmaki, and C. Sodegard, "Hyperfit: Hybrid media in personal nutrition and exercise management," in *Proceedings of the Second International Conference on Pervasive Computing Technologies for Healthcare*, Tampere, Finland, January 2008, pp. 222–226.

[28] "FoodScanner." [Online]. Available: http://www.dailyburn.com/foodscanner

[29] "Meal Snap," DailyBurn. [Online]. Available: http://www.dailyburn.com/apps

[30] "PhotoCalorie." [Online]. Available: http://www.photocalorie.com

[31] Tillett, "Image analysis for agricultural processes: A review of potential opportunities," *Journal of the Agricultural Engineering Research*, vol. 50, pp. 247–258, 1991.

[32] E. Parrish and A. Goksel, "Pictorial pattern recognition applied to fruit harvesting," *Transactions of the American Society Agricultural Engineers*, vol. 20, pp. 822–827, 1977.

[33] Whitaker, Miles, Mitchell, and Graultney, "Fruit location in a partially occluded image," *Transactions of the American Society Agricultural Engineers*, vol. 30, no. 3, pp. 591–597, 1987.

[34] D.-W. Sun and C.-J. Du, "Segmentation of complex food images by stick growing and merging algorithm," *Journal of Food Engineering*, vol. 61, pp. 17–26, 2004.

[35] Sites and Dewilche, "Computer vision to locate fruit on a tree," *Transactions of the American Society Agricultural Engineers*, vol. 31, no. 1, pp. 257–263, 1988.

[36] D. Mery and F. Pedreschi, "Segmentation of colour food images using a robust algorithm," *Journal of Food Engineering*, vol. 66, pp. 353–360, 2005.

[37] R. Harrel, D. Slaughter, and P. Adsit, "A fruit-tracking system for robotic harvesting," *Transactions of Machine Vision and Applications*, vol. 2, pp. 69–80, 1989.

[38] M. Puri, Z. Zhu, Q. Yu, A. Divakaran, and H. Sawhney, "Recognition and volume estimation of food intake using a mobile device," *Proceedings of the IEEE Workshop on Applications of Computer Vision*, December 2009.

[39] S. Arivazhagan, R. Newlin Shebiah, S. Selva Nidhyanandhan, and L. Ganesan, "Fruit recognition using color and texture features," *Journal of Emerging Trends in Computing and Information Sciences*, vol. 1, no. 2, pp. 90– 94, 2010.

[40] W. Qiu and S. Shearer, "Maturity assessment of broccoli harvest using the discrete fourier transform," *Transactions of the American Society Agricultural Engineers*, vol. 91, 1991.

[41] Y. Rui and T. S. Huang, "Image retrieval: Current techniques, promising directions, and open issues," *Journal of Visual Communication and Image Representation*, vol. 10, pp. 39–62, 1999.

[42] A. W. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349–1380, December 2000.

[43] S. Yang, M. Chen, D. Pomerleau, and R. Sukhankar, "Food recognition using statistics of pairwise local features," *Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2010.

[44] T. Joutou and K. Yanai, "A food image recognition system with multiple kernel learning," *Proceedings of International Conference on Image Processing (ICIP)*, October 2009.

[45] W. Wu and J. Yang, "Fast food recognition from videos of eating for calorie estimation," *Proceedings of the International Conference on Multimedia and Expo (ICME)*, pp. 1210–1213, June 2009.

[46] K. Kitamura, T. Yamasaki, and K. Aizawa, "Foodlog: Capture, analysis and retrieval of personal food images via web," *Proceedings of ACM multimedia workshop on Multimedia for cooking and eating activities*, November 2009.

[47] F. Kong and J. Tan, "Dietcam: Regular shape food recognition with a camera phone," *Proceedings of the International Conference on Body Sensor Networks*, pp. 127–132, May 2011.

[48] W. Zhang, B. Yu, G. Zelinsky, and D. Samaras, "Object class recognition using multiple layer boosting with heterogeneous features," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 66–73, 2005.

[49] J. Ponce, M. Hebert, C. Schmid, and A. Zisserman, *Toward Category-level Object Recognition*. Springer, 2006.

[50] A. Biem and S. Katagiri, "Feature extraction based on minimum classification error/generalized probabilistic descent method," *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 275–278, April 1993.

[51] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," *Proceedings of the International Workshop on Statistical Learning in Computer Vision*, 2004.

[52] G. Finlayson, M. Drew, and B. Funt, "Spectral sharpening: Sensor transformations for improved color constancy," *Journal Optical Society of America*, vol. 11, no. 5, p. 1553, 1994.

[53] van de Sande, K., T. Gevers, and C. Snoek, "Evaluating color descriptors for object and scene recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1582 – 1595, 2010.

[54] M. Fischler and R. Elschlager, "The representation and matching of pictorial structures," *IEEE Transactions on Computers*, vol. 1, no. 22, pp. 67 – 92, 1973.

[55] R. Fergus, P. Perona, and A. Zisserman, "Sparse object category model for efficient learning and exhaustive recognition," *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 380–387, 2005.

[56] V. Vapnik, "The nature of statistical learning theory," in *Springer-Verlag*, New York, NY, 1995.

[57] P. Somol and J. Novovicova, "Evaluating the stability of feature selectors that optimize feature subset cardinality," *Proceedings of the International Workshop Structural, Syntactic, and Statistical Pattern Recognition*, pp. 956 – 966, 2008.

[58] J. Fan and Y. Fan, "High dimensional classification using features annealed independence rules," *Annals of Statistics*, vol. 38, no. 6, pp. 2605 – 2637, 2007.

[59] M. Varma and D. Ray, "Learning the discriminative power-invariance trade-off," *Proceedings of the International Conference on Computer Vision (ICCV)*, 2007.

[60] A. Oliva and A. Torralba, "The role of context in object recognition." *Trends in Cognitive Sciences*, vol. 11, no. 12, pp. 520 – 527, 2007.

[61] C. Galleguillos and S. Belongie, "Context based object categorization: A critical survey," *Computer Vision and Image Understanding*, vol. 114, pp. 712–722, 2010.

[62] K. Murphy, A. Torralba, and W. Freeman, "Using the forest to see the trees: A graphical model relating features, objects, and scenes," in *Advances in Neural Information Processing Systems*, 2003.

[63] A. Torralba, K. Murphy, W. Freeman, and M. Rubin, "Context-based vision system for place and object recognition," *Proceedings of the International Conference on Computer Vision (ICCV)*, pp. 273–280, 2003.

[64] M. Zhu, "Multilevel image segmentation with application in dietary assessment and evaluation," Ph.D. dissertation, Purdue University, West Lafayette, IN, USA, 2011.

[65] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[66] F. Zhu, M. Bosch, N. Khanna, C. J. Boushey, and E. J. Delp, "Multilevel segmentation for food classification in dietary assessment," *Proceedings of the International Symposium on Image and Signal Processing and Analysis (ISPA)*, September 2011.

[67] F. Zhu, M. Bosch, T. Schap, N. Khanna, D. Ebert, C. J. Boushey, and E. J. Delp, "Segmentation assisted food classification for dietary assessment," *Proceedings of the IS&T/SPIE Conference on Computational Imaging IX*, vol. 7873, January 2011.

[68] C. Xu, N. Khanna, C. J. Boushey, and E. J. Delp, "Low complexity image quality measures for dietary assessment using mobile devices," *Proceedings of the IEEE International Symposium on Multimedia*, pp. 351–356, December 2011.

[69] C. Xu, F. Zhu, N. Khanna, C. J. Boushey, and E. J. Delp, "Image enhancement and quality measures for dietary assessment using mobile devices," *Proceedings of the IS&T/SPIE Conference on Computational Imaging X*, vol. 8296, January 2012.

[70] F. Zhu, M. Bosch, I. Woo, S. Kim, C. J. Boushey, D. S. Ebert, and E. J. Delp, "The use of mobile devices in aiding dietary assessment and evaluation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 4, pp. 756–766, August 2010.

[71] M. Bosch, F. Zhu, N. Khanna, C. J. Boushey, and E. J. Delp, "Food texture descriptors based on fractal and local gradient information," *Proceedings of the European Signal Processing Conference (Eusipco)*, September 2011.

[72] M. Bosch, F. Zhu, N. Khanna, C. J. Boushey, and E. J. Delp, "Combining global and local features for food identification and dietary assessment," *Proceedings of the International Conference on Image Processing (ICIP)*, 2011.

[73] I. Woo, K. Ostmo, S. Kim, D. S. Ebert, E. J. Delp, and C. J. Boushey, "Automatic portion estimation and visual refinement in mobile dietary assessment," *Proceedings of the IS&T/SPIE Conference on Computational Imaging VIII*, 2009.

[74] J. Chae, I. Woo, S. Kim, R. Maciejewski, F. Zhu, E. J. Delp, C. J. Boushey, and D. S. Ebert, "Volume estimation using food specific shape templates in mobile image-based dietary assessment," *Proceedings of the IS&T/SPIE Conference on Computational Imaging IX*, vol. 7873, 2011.

[75] K. Ostmo, "Automatic portion estimation and visual refinement in automatic portion estimation and visual refinement in mobile dietary assessment," Master's thesis, Purdue University, 2009.

[76] "USDA food and nutrient database for dietary studies, 1.0." Beltsville, MD: Agricultural Research Service, Food Surveys Research Group, 2004.

[77] "Stockfood - The Food Image Agency. Food pictures for professionals." [Online]. Available: http://www.stockfood.com

[78] "Food Testing Image Database," Appealing Products, Inc. [Online]. Available: http://appealingproducts.com/photobank/fpdk/

[79] "Food Photo Gallery by Food-Image.com." [Online]. Available: http://www.food{-}image.com

[80] I. Biederman, "A theory of human image understanding," *Psychological Review*, no. 96, pp. 115 – 147, 1987.

[81] A. Witkins, "Scale-space filtering," *Proceedings of the 8th International Joint Conference on Artificial Intelligence*, pp. 1019 – 1023, 1983.

[82] J. Koenderink, "The structure of images," *Biological Cybernetics*, no. 50, pp. 363 – 396, 1984.

[83] T. Lindeberg, *Scale-Space Theory in Computer Vision*. Kluwer Academic Publishers, 1994.

[84] L. Florack, "The syntactical structure of scalar images," Ph.D. dissertation, Universiteit Utrecht, Utrecht, Netherlands, 1993.

[85] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," *2006 International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2169 – 2176, 2004.

[86] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 27, no. 10, pp. 1615– 1630, 2005.

[87] B. Manjunath, P. Salembier, and T. Sikora, *Introduction to MPEG-7: Multimedia Content Description Interface*. Wiley and Sons, USA, 2002.

[88] G. Svaetichin, "Spectral response curves from single cones," *Acta Physiol Scand Suppl.*, vol. 134, pp. 17 – 46, 1956.

[89] CIE, "Commission internationale d'eclairage proceedings, 1931," Cambridge University Press, Cambridge, 1932.

[90] S. Shafer, "Using color to separate reflection components," *COLOR research and application*, vol. 10, no. 4, pp. 210 – 218, 1985.

[91] J. van de Weijer, T. Gevers, and A. Smeulders, "Robust photometric invariant features from the color tensor," *IEEE Transactions on Image Processing*, vol. 154, no. 1, pp. 118– 127, 2006.

[92] F. Mindru, T. Tuytelaars, L. Van Gool, and T. Moons, "Moment invariants for recognition under changing viewpoint and illumination," *Computer Vision and Image Understanding*, vol. 94, no. 1-3, pp. 3 – 27, 2004.

[93] G. Kennel, *Color and Mastering for Digital Cinema*. Focal Press, 2006.

[94] H. Fairman, M. Brill, and H. Hemmendinger, "How the cie 1931 color-matching functions were derived from wright-guild data," *Color Research and Application*, vol. 22, pp. 11 – 22, 1997.

[95] A. Ford and A. Roberts, "Color space conversions," August 1998.

[96] C. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, 623 – 656, July, October 1948.

[97] K. Popat and R. Picard, "Scale-space filtering," *Proceedings of the SPIE Visual Communications*, 1993.

[98] G. Derefeldt and T. Swartling, "Color concept retrieval by free color naming," *Displays*, vol. 16, pp. 69 – 77, 1995.

[99] J. Cheng, "Perceptually-based texture and color features for image segmentation and retrieval," Ph.D. dissertation, Northwestern University, Evanston, IL, 2003.

[100] W. Ma, Y. Deng, and B. Manjunath, "Tools for texture/color based search of images," *Proceedings of the SPIE Human Vision and Electronic Imaging II*, vol. 3016, 1997.

[101] M. Amadasun and R. King, "Textural features corresponding to textural properties," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 19, pp. 1264 – 1274, 1989.

[102] B. Julesz, "Textons, the elements of texture perception and their iteractions," *Nature*, vol. 290, pp. 91 – 97, 1981.

[103] A. Materka and M. Strzelecki, "Texture analysis methods - a review," Technical University of Lodz, Institute of Electronics, Tech. Rep., 1998.

[104] R. M. Haralick, "Statistical and structural approaches to texture," *Proceedings of the IEEE*, vol. 67, no. 5, pp. 786–804, May 1979.

[105] M. H. Bharati, J. J. Liu, and J. F. MacGregor, "Image texture analysis: methods and comparisons," *Chemometrics and Intelligent Laboratory Systems*, vol. 72, no. 1, pp. 57 – 71, 2004.

[106] I. Elfadel and R. Picard, "Gibbs random fields, cooccurrences, and texture modeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 16, no. 1, pp. 24 – 37, January 1994.

[107] A. Pentland, "Fractal-based description of natural scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 9, pp. 661 – 674, 1984.

[108] M. L. Comer and E. J. Delp, "Segmentation of textured images using a multiresolution gaussian autoregressive model," *IEEE Transactions on Image Processing*, vol. 8, pp. 408 – 420, March 1999.

[109] E. J. Delp, R. L. Kashyap, and O. Mitchell, "Image data compression using autoregressive time series models," *Pattern Recognition*, vol. 11, pp. 313–323, June 1979.

[110] M. Unser, "Texture classification and segmentation using wavelet frames," *IEEE Transactions on Image Processing*, vol. 4, pp. 1549–1560, November 1995.

[111] A. Bovik, M. Clark, and W. Geisler, "Multichannel texture analysis using localized spatial filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 12, pp. 55–73, 1990.

[112] A. Jain and F. Farrokhnia, "Unsupervised texture segmentation using gabor filters," *Pattern recognition*, vol. 24, no. 12, pp. 1167–1186, 1991.

[113] H. Voorhees and T. Poggio, "Detecting textons and texture boundaries in natural images," *Proceedings of the First International Conference on Computer Vision*, pp. 250 – 258, 1987.

[114] K. Laws, "Rapid texture identification," *Proceedings of SPIE Image Processing for Missile Guidance*, vol. 238, pp. 376 – 380, 1980.

[115] S. Lazebnik, C. Schmid, and J. Ponce, "A sparse texture representation using affine-invariant regions," *Proceedings of the International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 319 – 324, 2003.

[116] M. Varma and A. Zisserman, "Classifying images of materials: Achieving viewpoint and illumination independence," *Proceedings of the European Conference on Computer Vision (ECCV)*, vol. 3, pp. 255 – 271, 2002.

[117] R. Haralick, K. Shanmugan, and I. Dinstein, "Textural feature for image classification," *IEEE Transactions on Systems, man and cybernetics*, vol. SMC-3, no. 6, pp. 610–621, November 1973.

[118] B. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Transactions On Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837–842, August 1996.

[119] Y. Xu, X. Yang, H. Ling, and H. Ji, "A new texture descriptor using multifractal analysis in multi-orientation wavelet pyramid," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 161–168, 2010.

[120] F. Argenti, L. Alparone, and G. Benelli, "Fast algorithms for texture analysis using co-occurrence matrices," *Proceedings of the Radar and Signal Processing Conference*, vol. 137, pp. 443–448, December 1990.

[121] D. Clausi and M. Jernigan, "A fast method to determine cooccurrence texture features and its applicability to image interpretation," *Proceedings of the 26th Symposium on Remote Sensing of Environment and 18th Annual Symposium of the Canadian Remote Sensing*, March 1995.

[122] H. Farid and E. Simoncelli, "Differentiation of discrete multi-dimensional signals," *IEEE Transactions on Image Processing*, vol. 13, pp. 496–508, 2004.

[123] H. Bay, A. Ess, and L. Tuytelaars, T. Van Gool, "Surf: Speeded up robust features," *Proceedings of the International Conference on Computer Vision and Image Understanding (CVIU)*, vol. 110, no. 3, pp. 346– 359, 2008.

[124] L. Richardson, "The problem of contiguity: An appendix of statistics of deadly quarrels," *General Systems Yearbook*, vol. 6, pp. 139 – 187, 1961.

[125] B. Mandelbrot, *Fractals: Form, Chance and Dimension.* Freeman, San Francisco, CA, 1977.

[126] P. Sailhac and F. Seyler, *Texture Characterisation of ERS-1 Images by Regional Multifractal Analysis.* Levy Vehel, Fractals in Engineering, Springer, 1977.

[127] K. Falconer, *Fractal Geometry: Mathematical Foundations and Applications.* Wiley, England, 1990.

[128] Y. Xu, H. Ji, and C. Fermuller, "Viewpoint invariant texture description using fractal analysis," *International Journal of Computer Vision*, vol. 83, no. 1, pp. 85 – 100, 2009.

[129] T. Kadir and M. Brady, "Scale, saliency and image description," *International Journal of Computer Vision*, vol. 45, no. 2, pp. 83–105, 2001.

[130] M. Foreman, S. Kechris, A. Louveau, and B. Weiss, *Descriptive Set Theory and Dynamical Systems.* London Mathematical Society Lecture Note Series, 2000.

[131] N. Sarkar and B. Chaudhuri, "An efficient differential box-counting approach to compute fractal dimension of images," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 2, pp. 115 – 120, 1994.

[132] S. Grigorescu, N. Petkov, and P. Kruizinga, "Comparison of texture features based on gabor filters," *IEEE Transactions on Image Processing*, vol. 11, no. 10, pp. 1160–1167, October 2002.

[133] C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition," *IEEE Transactions on Image Processing*, vol. 11, no. 4, pp. 467–476, April 2002.

[134] P. Kruizinga, N. Petkov, and S. E. Grigorescu, "Comparison of texture features based on gabor filters," *Proceedings of the 10th International Conference on Image Analysis and Processing*, p. 142, September 1999.

[135] A. Bosch, A. Zisserman, and X. Munoz, "Scene classification using a hybrid generative/discriminative approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 4, pp. 712 – 727, 2008.

[136] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[137] E. Nowak, F. Jurie, and B. Triggs, "Sampling strategies for bag-of-features image classification," *Proceedings of the European Conference on Computer Vision (ECCV)*, 2006.

[138] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 2, no. 60, pp. 91–110, 2004.

[139] W. Forstner, "A feature-based correspondence algorithm for image matching," *International Architecture Photogrammetry & Remote Sensing*, vol. 26, no. 3, pp. 150–166, 1986.

[140] C. Harris and M. Stephens, "A combined corner and edge detector," *Proceedings of the 4th Alvey Vision Conference*, pp. 147 – 151, 1988.

[141] K. Mikolajczyk and C. Schmid, "Scale and affine invariant interest point detectors," *International Journal of Computer Vision*, vol. 1, no. 60, pp. 63 – 86, 2004.

[142] J. Babaud, A. Witkin, M. Baudin, and R. Duda, "Uniqueness of the gaussian kernel for scale-space filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 8, no. 1, pp. 26–33, 1986.

[143] S. Gilles, "Robust description and matching of images," Ph.D. dissertation, University of Oxford, Oxford, England, 1998.

[144] H. Tamura, S. Mori, and T. Yamawaki, "Textural features corresponding to visual perception," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 8, pp. 530 – 534, 1978.

[145] W. Freeman and Y. Adelson, "The design and use of steerable filters," *IEEE Transactions on System, Man, and Cybernetics*, pp. 460 – 473, 1978.

[146] E. Tola, V. Lepetit, and P. Fua, "Daisy: An efficient dense descriptor applied to wide-baseline stereo," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 32, no. 5, pp. 815 – 830, 2005.

[147] T. Mitchell, *Machine Learning*. The McGraw-Hill Companies, Inc., 1997.

[148] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," *Proceedings of the 23rd International Conference on Machine Learning*, 2006.

[149] Z. Ghahramani, *Unsupervised Learning*. Springer-Verlag, 2004.

[150] L. Kaelbling, M. Littman, and A. Moore, "Reinforcement learning: a survey," *Journal of Artificial Intelligence Research*, vol. 4, pp. 237 – 285, 1996.

[151] I. Rezek, D. Leslie, S. Reece, S. Roberts, A. Rogers, R. Dash, and N. Jennings, "On similarities between inference in game theory and machine learning," *Journal of Artificial Intelligence Research*, vol. 33, pp. 259 – 283, 2008.

[152] R. Duda and P. Hart, *Pattern classification and scene analysis.* John Wiley and Sons, New York, 1973.

[153] R. Bellman, *Adaptive control processes: a guided tour.* Princeton University Press, 1961.

[154] J. Friedman, "Another approach to polychotomous classification," Technical report, Department of Statistics, Stanford University, 1996.

[155] C. Hsu and C. Lin, "A comparison of methods for multi-class support vector machines," *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 415–425, 2002.

[156] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998.

[157] S. Canu, Y. Grandvalet, V. Guigue, and A. Rakotomamonjy, "Svm and kernel methods matlab toolbox," Perception Systmes et Information, INSA de Rouen, Rouen, France, 2005.

[158] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid, "Local features and kernels for classification of texture and object categories: An in-depth study," 2005, INRIA Technical Report RR-5737.

[159] O. Duda, P. Hart, and D. Stork, *Pattern Classification.* John Wiley & Sons, 2000.

[160] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," *Proceedings of the Computer Vision and Pattern Recognition (CVPR)*, pp. 2161 – 2168, 2006.

[161] K. Mikolajczyk, B. Leibe, and C. Schiele, "Multiple object class detection with a generative model," *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 26– 436, 2006.

[162] B. Ommer and J. Buchmann, "Learning the compositional nature of visual object categories for recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 32, no. 3, pp. 501 – 516, 2010.

[163] J. V. Gemert, A. S. C.J. Veenman, and J. Geusebroek, "Visual word ambiguity," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 32, no. 7, pp. 1271 – 1283, 2010.

[164] H. Jegou, C. Schmid, H. Harzallah, and J. Verbeek, "Accurate image search using the contextual dissimilarity measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 1– 11, 2010.

[165] I. Jolliffe, *Principal Component Analysis.* Springer-Verlag, 1986.

[166] B. Scholkopf, A. Smola, and K. Muller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, no. 5, pp. 1299 – 1319, 1998.

[167] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *Proceedings of the Neural Information Processing Systems*, 2001.

[168] R. Jenssen, "Kernel entropy component analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 5, pp. 847 – 860, 2010.

[169] L. Xu, A. Krzyzak, and C. Suen, "Methods of combining multiple classifiers and their applications to handwriting recognition," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, no. 3, pp. 418 – 435, 1992.

[170] I. Biederman, R. Mezzanotte, and J. Rabinowitz, "Scene perception: detecting and judging objects undergoing relational violations," *Cognitve Psychology*, vol. 14, no. 2, pp. 143 – 177, 1982.

[171] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie, "Objects in context," *Proceedings of the International Conference on Computer Vision (ICCV)*, 2007.

[172] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," *Proceedings of the International Conference on Machine Learning (ICML)*, 2001.

[173] C. Sutton and A. McCallum, *An Introduction to conditional random fields for relational learning.* L. Getoor and B. Taskar, editors, Introduction to Statistical Relational Learning, MIT Press, 2006.

[174] D. Liu and J. Nocedal, "On the limited memory bfgs method for large scale optimization," *Mathematical Programming*, vol. 45, pp. 503 – 528, 1989.

[175] K. Murphy and M. Schmidt, "Crf toolbox," http://www.cs.ubc.ca/ murphyk/Software/CRF/crf.html.

[176] P. Brodatz, *Textures: A Photographic Album for Artists and Designers.* Dover, NewYork, 1966.

[177] S. Lazebnik, C. Schmid, and J. Ponce, "A sparse texture representation using local affine regions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, pp. 1265 – 1278, 2005.

[178] Y. Xu, H. Ling, and C. Fermuller, "A projective invariant for textures," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 1932–1939, 2006.

[179] K. Mikolajczyk, T. Tuytelaars, C. Schmid, J. Zisserman, A. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *International Journal on Computer Vision*, vol. 65, no. 1, pp. 43 – 72, 2005.

[180] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal on Computer Vision*, vol. 88, pp. 303–338, 2010.

[181] M. Gregory and Y. Altun, "Using conditional random fields to predict pitch accents in conversational speech," *Proceedings of the 42nd Meeting of the Association for Computational Linguistics*, pp. 677 – 683, 2004.

[182] D. Olson and D. Delen, *Advanced Data Mining Techniques*. New York, NY: Springer, 2008.

[183] H. Korth, A. Silberschatz, and S. Sudarshan, *Database System Concepts*. New York, NY: McGraw-Hill, 1997.

[184] Japan Electronic Industry Development Association (JEIDA), "Design rule for camera file system, version 1.0." 1998.

[185] N. Khanna, H. Eicher-Miller, C. J. Boushey, S. B. Gelfand, and E. J. Delp, "Temporal dietary patterns using kernel k-means clustering," *Proceeding of the IEEE International Symposium on Multimedia (ISM)*, pp. 375–380, December 2011.

[186] J. Bormans, J. Gelissen, and A. Perkis, "Mpeg-21: The 21st century multimedia framework," *IEEE Signal Processing Magazine*, vol. 20, pp. 53–62, 2003.

[187] International Organization for Standardization, "Information and Documentation. The Dublin Core metadata element set," Brittish Standard, 2009.

[188] S. Stella, S. Kelkar, and M. Okos, "Predicting and 3-d laser scanning for determination of apparent density of porous food," in *IFT Annual Meeting and Food Expo*, July 2010.

[189] S. Kelkar, S. Stella, and M. Okos, "X-ray micro computed tomography (ct): A novel method to measure density of porous food," in *IFT Annual Meeting and Food Expo*, July 2010.

[190] R. Palmer, *The Bar-Code Book*. Helmers Publishing, 1989.

[191] S. Kim, T. Schap, M. Bosch, R. Maciejewski, E. J. Delp, D. S. Ebert, and C. J. Boushey, "A mobile user interface for image-based dietary assessment," *Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia*, December 2010.

[192] B. Momjan, *PostgreSQL: Introduction and Concepts*. Reading, MA: Pearson Education, 2000.

[193] T. Schap, B. Six, E. J. Delp, D. S. Ebert, D. Kerr, and C. J. Boushey, "Adolescents in the united states can identify familiar foods at the time of consumption and when prompted with an image 14 h postprandial, but poorly estimate portions," *Public Health Nutrition*, vol. 14, no. 07, pp. 1184–1191, February 2011.

[194] B. Six, T. Schap, F. Zhu, A. Mariappan, M. Bosch, E. J. Delp, D. S. Ebert, D. Kerr, and C. J. Boushey, "Evidence-based development of a mobile telephone food record," *Journal of American Dietetic Association*, pp. 74–79, January 2010.

[195] N. Khanna, C. J. Boushey, D. Kerr, M. Okos, D. S. Ebert, and E. J. Delp, "An overview of the technology assisted dietary assessment project at purdue university," *Proceedings of the IEEE International Symposium on Multimedia*, pp. 290–295, December 2010.

[196] C. Aulds, *Linux Apache web server administration*. Wiley, England, 2002.

APPENDIX

A. DATABASE SOFTWARE

We have integrated a web-based interface into the TADA system to provide dietitians, engineers, and researchers with a wide range of information including, nutritional food information, participant statistics, image analysis evaluation performances, and other. Typically there are three main elements in any web application architecture: a relational database, middleware, and web server.

The TADA databases were developed using the PostgreSQL language, based on a SQL syntax [192]. The middleware consists of a class of programming modules that work closely with the web server to interpret the requests made from the web interface application, interact with other programs on the server to fulfill the requests, and then, indicate to the web server exactly what resources to serve to the web interface application in the internet browser. This was developed using **web.py**[1]; a web framework for python that allows to create websites with SQL support. As part of the middleware we used as the Web Server Gateway Interface (WSGI) **flup**[2] to be the interface between the web server and the web interface application. The web interface application is supported by a python client module known as **psycopg2**[3]. As the web server we considered the open source software **Apache** web server for the web page front-end delivery content tool [196]. Due to the large size of our databases, we are using connection pools so that each database has a cache to maintain the data requests so they can be reused in the future. It also enhances the performance of executing the query commands from the web page by reutilizing the connections between the web-based interface and the SQL database. **Cheetah**[4] is used as the template engine for the web application.

[1]http://webpy.org/
[2]http://docs.python.org/howto/webservers.html
[3]http://initd.org/psycopg/
[4]http://www.cheetahtemplate.org/

VITA

VITA

Marc Bosch Ruiz was born in Barcelona, Spain. He obtained the Telecommunications Engineering Degree (B.S. + M.S.) from the Technical University of Catalonia (UPC) in 2006. He was a visitor scholar and researcher in the Video and Image Processing Laboratory at Purdue University between 2006 and 2007. Marc obtained his M.S. in Electrical and Computer Engineering from Purdue University, West Lafayette in 2009. He joined the PhD program at Purdue University in 2009. Since 2007, he has served as a Research Assistant in the Video and Image Processing Laboratory (VIPER).

His thesis advisor, Professor Edward J. Delp, is the Charles William Harrison Distinguished Professor of Electrical and Computer Engineering. While in the graduate program, he worked on projects sponsored by the National Institutes of Health (NIH). In 2007 he received the Archimedes Award for the best undergraduate engineering thesis from the Science and Education Ministry of Spain. He received the Meritorious New Investigator Award at the 2010 mHealth Summit.

His current research interests include image and video processing, image analysis, video coding, machine learning, computer vision and medical imaging.

He is a student member of the IEEE and the IEEE Signal Processing Society.

Marc Bosch's publications are:

Book Chapter:

1. Limin Liu, Fengqing Zhu, **Marc Bosch**, and Edward J. Delp, "Recent Advances in Video Compression: What's next?," book chapter in *Statistical Science and Interdisciplinary Research: Pattern Recognition, Image Processing and Video Processing*, Ed. Bhabatosh Chanda et. al., World Scientific Press, 2007.

Journal Articles:

1. **Marc Bosch**, Nitin Khanna, Carol J. Boushey, and Edward J. Delp, 'An Integrated Image-Based Food Database System with Application in Dietary Assessment," *IEEE Transactions on Information Technology in Biomedicine*, submitted.

2. Fengqing Zhu, **Marc Bosch**, Nitin Khanna, Carol J. Boushey and Edward J. Delp, "Multiple Hypothesis Image Segmentation and Classification with Application to Dietary Assessment," *IEEE Transactions on Image Processing*, submitted.

3. **Marc Bosch**, Fengqing Zhu, and Edward J. Delp, "Segmentation Based Video Compression Using Texture and Motion Models," *IEEE Journal of Selected Topics in Signal Processing*, November 2011. *to appear.*

4. Fengqing Zhu, **Marc Bosch**, Insoo Woo, SungYe Kim, Carol J. Boushey, David S. Ebert, Edward J. Delp, "The Use of Mobile Devices in Aiding Dietary Assessment and Evaluation," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 4, pp. 756-766, August 2010.

5. Bethany L. Six, TusaRebecca E. Schap, Fengqing Zhu, Anand Mariappan, **Marc Bosch**, Edward J. Delp, David S. Ebert, Deborah A. Kerr, Carol J. Boushey, "Evidence-Based Development of a Mobile Telephone Food Record," Journal of American Dietetic Association, January 2010, pp. 74-79.

Conference Papers

1. Fengqing Zhu, **Marc Bosch**, Ziad Ahmad, Nitin Khanna, Carol J. Boushey, Edward J. Delp, "Challenges in Using a Mobile Device Food Record Among Adults in Freeliving Situations," *mHealth Summit*, December, 2011, Washington D.C.

2. Meilin Yang, Ye He, Fengqing Zhu, **Marc Bosch**, Mary Comer, and Edward J. Delp, "Video Coding: Death Is Not Near", Proceedings of the 53rd International Symposium ELMAR, September 2011, Zadar, Croatia.

3. **Marc Bosch**, Fengqing Zhu, Nitin Khanna, Carol J. Boushey, Edward J. Delp, "Combining Global and Local Features for Food Identification and Dietary Assessment," *Proceedings of the IEEE International Conference on Image Processing*, Brussels, Belgium, September 2011.

4. Fengqing Zhu, **Marc Bosch**, Nitin Khanna, Carol J. Boushey, Edward J. Delp, "Multilevel Segmentation for Food Classification in Dietary Assessment," *Proceedings of the International Symposium on Image and Signal Processing and Analysis*, Dubrovnik, Croatia, September 2011.

5. **Marc Bosch**, Fengqing Zhu, Nitin Khanna, Carol J. Boushey, Edward J. Delp, "Food Texture Descriptors Based on Fractal and Local Gradient Information," *Proceedings of the European Signal Processing Conference (Eusipco)*, Barcelona, Spain, August 2011.

6. **Marc Bosch**, TusaRebecca E. Schap, Nitin Khanna, Fengqing Zhu, Carol J. Boushey, Edward J. Delp, "Integrated Databases for Mobile Dietary Assessment and Analysis," *Proceedings of the 1st IEEE International Workshop on Multimedia Services and Technologies for E-Health in conjunction with the International Conference on Multimedia and Expo (ICME)*, Barcelona, Spain, July 2011.

7. Fengqing Zhu, **Marc Bosch**, Nitin Khanna, TusaRebecca E. Schap, Carol J. Boushey, David S. Ebert, Edward J. Delp, "Segmentation Assisted Food Classification for Dietary Assessment", *Proceedings of Computational Imaging IX, IS&T/SPIE Electronic Imaging*, San Francisco, CA, January 2011.

8. SungYe Kim, TusaRebecca E. Schap, **Marc Bosch**, Ross Maciejewski, Edward J. Delp, David S. Ebert, and Carol J. Boushey, "A Mobile User Interface for Image-based Dietary Assessment", *Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia*, Limassol, Cyprus, December 2010.

9. **Marc Bosch**, Fengqing Zhu, TusaRebecca E. Schap, Carol J. Boushey, Deborah Kerr, Nitin Khanna, Edward J. Delp, "An Integrated Image- Based Food Database System with Application in Dietary Assessment," *mHealth Summit*, November, 2010, Washington D.C. *(Meritorious New Investigator Award)*

10. Nitin Khanna, Fengqing Zhu, **Marc Bosch**, Meilin Yang, Mary Comer and Edward J. Delp, "Information Theory Inspired Video Coding Methods: Truth Is Sometimes Better Than Fiction," Proceedings of the Third Workshop on Information Theoretic Methods in Science and Engineering, Tampere, Finland, August 16 - 18, 2010.

11. Fengqing Zhu, **Marc Bosch**, Carol J. Boushey, and Edward J. Delp, "An Image Analysis System for Dietary Assessment and Evaluation," *Proceedings of the IEEE International Conference on Image Processing*, September, 20010, Hong Kong.

12. **Marc Bosch**, Fengqing Zhu, Edward J. Delp, Perceptual Quality Evaluation for Texture and Motion Based Video Coding, Proceedings of the ICIP, IEEE International Conference on Image Processing, November 2009, Cairo, Egypt.

13. **Marc Bosch**, Fengqing Zhu, and Edward J. Delp, "An Overview of Texture and Motion based Video Coding at Purdue University," Proceedings of the 27th Picture Coding Symposium, Chicago, Illinois, May 6- 8, 2009.

14. Anand Mariappan, **Marc Bosch**, Fengqing Zhu, Carol J. Boushey, Deborah A. Kerr, David S. Ebert, Edward J. Delp, "Personal Dietary Assessment Using Mobile Devices," *Proceedings of the IS&T/SPIE Conference on Computational Imaging VII*, Vol. 7246, San Jose, January 2009.

15. **Marc Bosch**, Fengqing Zhu, and Edward J. Delp, Video Coding Using Motion Classification, Proceedings of the ICIP, IEEE International Conference on Image Processing, October 12-15, 2008, San Diego, CA.

16. **Marc Bosch**, Fengqing Zhu, and Edward J. Delp, "Models for texture based video coding," Proceedings of LNLA, IEEE International Workshop on Local and Non-Local Approximation in Image Processing, Lausanne, Switzerland, August 23-24 2008.

17. **Marc Bosch**, Fengqing Zhu, and Edward J. Delp, Spatial Texture Models for Video Compression, Proceedings of the ICIP, IEEE International Conference on Image Processing, September 16-19, 2007, San Antonio, TX.

18. Limin Liu, **Marc Bosch**, Fengqing Zhu, and Edward J. Delp, Recent advances in video compression: whats next?, Proceedings of the IEEE International Symposium on Signal Processing and its Applications (ISSPA 2007), February 12-15, 2007, Sharjah, United Arab Emirates (Plenary Paper).

19. **Marc Bosch**, Montse Najar, Unscented Kalman Filter for location in non-line-of-sight, in 14th European Signal Processing Conference (EUSIPCO), September 2006, Florence, Italy.