

CONTEXT BASED TATTOO IMAGE ANALYSIS
WITH APPLICATIONS IN PUBLIC SAFETY

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Joonsoo Kim

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 2017

Purdue University

West Lafayette, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF DISSERTATION APPROVAL

Dr. Edward J. Delp, Chair

School of Electrical and Computer Engineering

Dr. Mary L. Comer

School of Electrical and Computer Engineering

Dr. Mireille Boutin

School of Electrical and Computer Engineering

Dr. Zygmunt Pizlo

Department of Psychological Sciences

Approved by:

Dr. Venkataramanan Balakrishnan

Head of the School Graduate Program

This thesis is dedicated to my lovely wife, my lovely daughter, my parents, and my
parents in law.

ACKNOWLEDGMENTS

First of all, I would like to express my sincere gratitude and respect to my major advisor, Professor Edward J. Delp. He has offered many opportunities for my research and I have learnt lots of thing from that. I learnt how to start to solve my problem and how to proceed it until reaching the goal. Also, his advice and guidance helped me become an independent researcher and developer. I cannot express my gratitude toward him using one word. I feel very proud to have worked with him.

I am also grateful to all the members of my dissertation committee: Professor Zygmunt Pizlo, Professor Mary Comer, and Professor Mireille Boutin for their advice, guidance, and encouragement.

I am also grateful to Purdue University for providing me happy and unforgettable life in West Lafayette.

I would like to extend my gratitude to all my former and current brilliant colleagues in the VIPER laboratory, Dr. Chang Xu, Dr. Ye He, Dr. Albert Parra Pozo, Dr. Bin Zhao, Dr. Neeraj Gadgil, Jiaju Yue, Blance Delgado, Khalid Tahboub, Soonam Lee, He Li, Yu Wang, Dahjung Chung, David Joon Ho, Jeehyun Choe, Di Chen, Qingshuang Chen, Yuhao Chen, Shaobo Fang, Chichen Fu, David Gera, Shuo Han, Chang Liu, Daniel Mas, Javier Ribera, and Ruiting Shao. All of them have achieved really good works on their research and they have provided nice research atmosphere in the lab.

I also want to show my gratitude to all the members in our “G6 group”: Soonam Lee, Chung Hwan Kim, Kyubyung Kang, Jihwan Oh, and Haejun Chung. We joined to Purdue in 2012 together and have shared great memory so far. I believe that all of us will achieve great success after our graduation.

There is another group that I need to appreciate, the third group of church: Sangchoul Yi, Sungja Yoo, Junghee Min, Bodam Lee, Joonyup Eun, Joonho Lee, Jungsoon

Kang, Sunghwan Hwang, Hyunju Lee, Sukwon Lee, Wonjung Jang, Byungju Lee, and Jeehye Lee. From all the members in this group, I have learnt so many things: happiness, endurance, collaboration, etc. I really want to keep the relationship with these members after a graduation.

I would like to thank to Prof. Yoonsik Choe. When I was a Master student, he offered many opportunities and chances for me to achieve. Through the chances, I could learn lots of things. Also, his teaching and guidance made me to do research independently, I want to show my respect and gratitude to him as well.

I would like to thank to my parents and my parents in law for supporting and trusting me. I cannot tell you how much they have loved and supported me and my family. With their support and trust, I could have focused on my research more. Without their support, I cannot imagine the current moment that I am writing this thesis. I would make it up to them forever.

There is another important person who always prayed for me and loved me. She is my grandmother, who passed away a few months ago. I cannot express her love to me with any words in the world. If she is still alive, I would have showed this thesis to her. I am very sad because I cannot do it.

Last, I really want to say “thank you and love you” to my lovely wife, Gyunwook Kim. She always has been with me and she encouraged me whenever I had hard time. Since she is always with me, I can have an opportunity to write this thesis. The time and memory I have shared with my wife in Purdue will unforgettable forever. Also, I want to say “Thanks for your birth from us and I love you” to my lovely daughter, Hasun Kim.

The tattoo images used in this thesis were obtained in cooperation with the Indiana State Police as part of the INGang Network. I gratefully acknowledge their cooperation. This work was supported by the U.S. Department of Homeland Security’s VACCINE Center under Award Number 2009-ST-061-CI0001.

TABLE OF CONTENTS

	Page
LIST OF TABLES	x
LIST OF FIGURES	xi
ABSTRACT	xiv
1 INTRODUCTION	1
1.1 Tattoo Segmentation	1
1.2 Tattoo Image Retrieval Based on Image Matching	3
1.3 Tattoo Image Classification Based On Sparse Coding	6
1.4 Contributions Of The Thesis	8
1.5 Publications Resulting From Our Work	12
2 TATTOO SEGMENTATION	14
2.1 Review of Existing Methods	14
2.2 Efficient Graph-Cut Tattoo Segmentation	18
2.2.1 Detection of Possible Segmentation Regions	19
2.2.2 Graph-Cut Segmentation	20
2.2.3 Post-Processing	24
2.3 Efficient Graph-Cut Tattoo Segmentation With Body Boundary Removal	24
2.3.1 Body Boundary Removal (BBR)	25
2.4 Experimental Results	28
2.4.1 Tattoo Segmentation	28
2.4.2 Tattoo Localization	29
2.5 Conclusions and Future Work	32
3 TATTOO IMAGE RETRIEVAL BASED ON IMAGE MATCHING	36
3.1 Review of Existing Methods	36

	Page
3.2 Tattoo Image Retrieval System Based On Multiple Histograms Based Local Context (MHLC) Descriptor	42
3.2.1 System Overview	42
3.2.2 Multiple Different Sized-Bin Histograms based Local Context (MHLC) Descriptor	43
3.2.3 Global Shape Descriptor	46
3.2.4 Image Matching	47
3.2.5 Experimental Results	49
3.3 Tattoo Image Retrieval System Based On Dense Multiple Histograms Based Local Context (DMHLC) Descriptor	57
3.3.1 System Overview	57
3.3.2 Dense Multiple Different Sized-Bin Histograms based Local Context (DMHLC) Descriptor	58
3.3.3 Image Matching	61
3.3.4 Experimental Results	62
3.4 Tattoo Image Retrieval System Based On Inductive Matching	65
3.4.1 System Overview	67
3.4.2 The Modified Inductive Matching	68
3.4.3 Experimental Results	69
3.5 Tattoo Image Retrieval For Tattoo Identification	75
3.5.1 System Overview	76
3.5.2 MHLC Descriptor	77
3.5.3 Image Matching	78
3.5.4 Experimental Results	78
3.6 Tattoo Image Retrieval For Region of Interest	81
3.6.1 System Overview	82
3.6.2 Scale Invariant Feature Transform	83
3.6.3 Local Self Similarity	84
3.6.4 Image Descriptor	85

	Page
3.6.5 Weighted Distance Similarity For Image Matching	85
3.6.6 Experimental Results	86
3.7 Conclusions and Future Work	94
4 TATTOO IMAGE CLASSIFICATION BASED ON SPARSE CODING	96
4.1 Review of Existing Methods	96
4.2 Center-Aligned Spatial Pyramid (CASP) Based Object Classification	99
4.2.1 Sparse Coding and Max Pooling	100
4.2.2 Center-Aligned Spatial Pyramid Generation	100
4.2.3 Image Descriptor Robust to Object Deformation	103
4.3 Experimental Results	104
4.3.1 Caltech-101 Dataset	105
4.3.2 Caltech-256 Dataset	105
4.3.3 Evil Tattoo Dataset	107
4.3.4 Discussion	107
4.4 Conclusions and Future Work	108
5 SHAPE MATCHING AND RETRIEVAL USING A SELF SIMILAR AFFINE INVARIANT DESCRIPTOR	109
5.1 Review of Existing Shape Methods	110
5.2 Proposed Shape Retrieval System	111
5.2.1 System Overview	111
5.2.2 Self Similar Affine Invariant (SSAI) Descriptor	111
5.2.3 Multiple SSAI Descriptors/Multiple Levels	115
5.2.4 Image Matching	116
5.3 Experimental Results	117
5.4 Conclusions and Future Work	119
6 SYSTEM IMPLEMENTATION	123
6.1 Overall System	123
6.2 Mobile Application	124

	Page
6.2.1 Browse Image	125
6.2.2 Browse Database	125
6.2.3 Capture Image	128
6.2.4 Send to Server	129
6.2.5 Find Similar Images	130
6.2.6 Setting	131
6.3 Web Interface System	131
6.3.1 Browse Database	134
6.3.2 Find Similar Images	135
7 CONCLUSIONS	139
7.1 Summary	139
7.2 Future Work	142
7.3 Publications Resulting From Our Work	143
REFERENCES	145
VITA	159

LIST OF TABLES

Table	Page
2.1 Recall and Accuracy	31
3.1 The description for our datasets	50
3.2 CMC in dataset 1 (<i>unit</i> : %)	51
3.3 CMC in dataset 2 (<i>unit</i> : %)	55
3.4 CMC in dataset 3 (<i>unit</i> : %)	56
3.5 CMC in dataset 1 (<i>unit</i> : %)	63
3.6 CMC in dataset 2 (<i>unit</i> : %)	65
3.7 CMC using DMHLC with robust image similarity (<i>unit</i> : %)	70
3.8 CMC using DMHLC With image similarity in [21] (<i>unit</i> : %)	71
3.9 CMC using MHLC With robust image similarity (<i>unit</i> : %)	73
3.10 CMC using MHLC with image similarity in [21] (<i>unit</i> : %)	74
3.11 The description for our datasets	78
3.12 CMC in dataset 4 (<i>unit</i> : %)	79
3.13 CMC and MAP in TID dataset with background images as reported in [11]	80
3.14 The description for our datasets	87
3.15 CMC and MAP for the ROI dataset without background images	89
3.16 CMC and MAP in ROI dataset with background image	89
3.17 CMC and MAP in ROI dataset with background images as reported in [11]	91
4.1 Classification accuracy (%) on Caltech-101 and Caltech-256 datasets . .	106
4.2 Classification accuracy (%) on Evil Tattoo dataset	107
5.1 Bull's Eye Score - MPEG-7 dataset	120
5.2 Retrieval result - Articulated dataset	121

LIST OF FIGURES

Figure	Page
1.1 The Convolutional neural network (CNN) architecture	3
1.2 Example of the BOW model with the spatial pyramid	7
2.1 The SegNet architecture	16
2.2 Our proposed tattoo image segmentation system	19
2.3 Overall segmentation process	20
2.4 Examples of false contours: In Figure 2.4(b), the red rectangular box is the minimum bounding box for a tattoo region, and the blue rectangular boxes are the minimum bounding boxes which include false contours. Using additional constraint, Equation (2.9), a tattoo region is only segmented as depicted in Figure 2.4(c)	23
2.5 Our tattoo segmentation system with BBR	26
2.6 Graph cut segmentation with BBR	27
2.7 Incorrect tattoo image segmentation	29
2.8 Tattoo image segmentation results	33
2.9 Tattoo image segmentation results	34
2.10 Tattoo image segmentation results	35
3.1 General image retrieval system based on image matching	37
3.2 The Siamese neural network architecture	38
3.3 Tattoo image retrieval system based on MHLC descriptor	43
3.4 Example of computing θ'_c and the polar histogram: In (a) the orientation with maximum feature location count is $\theta'_c = \pi/18$. In (d) features are rotated $8\pi/18$ with respect to (a), yielding $\theta'_c = 2\pi/18$. (b) and (e) show the aligned feature locations. The histograms computed in (c) and (f) have the same count distribution, but with a circular displacement	48
3.5 Sample tattoo images in this dataset	51
3.6 Retrieval performance comparison of our proposed methods, [4], SIFT+ [117], [120], and [21] using CMC in Dataset 1	52

Figure	Page
3.7 Retrieval performance comparison of our proposed methods, [4], SIFT+ [117], [120], and [21] using CMC in Dataset 2	54
3.8 Tattoo image retrieval system based on DMHLC descriptors	58
3.9 Example of the DMHLC descriptor generation	59
3.10 Retrieval performance comparison of our proposed methods, RIS [4], SIFT+SC [117], SIFT+GC [120], and SIFT [21] using CMC in Dataset 1	62
3.11 Retrieval performance comparison of our proposed methods, RIS [4], SIFT+SC [117], SIFT+GC [120], and SIFT [21] using CMC in dataset 2	64
3.12 Example of the diffusion process: In 3.12a the black dots are the database images that belong to two different classes and blue and red dots are input images. The diffusion process can cluster them into two class correctly . .	67
3.13 Tattoo image retrieval system based on Modified Inductive Matching . . .	68
3.14 CMC curve using DMHLC with robust image similarity	71
3.15 CMC curve using DMHLC with image similarity in [21]	72
3.16 CMC curve using MHLC with robust image similarity	73
3.17 CMC curve using MHLC with image similarity in [21]	74
3.18 Tattoo image retrieval system for tattoo identification	77
3.19 Retrieval performance of our proposed method and MSU [4] using CMC in TID dataset with background image	81
3.20 Retrieval performance of our proposed method and MSU [4] using CMC in TID dataset without background image	82
3.21 Tattoo image retrieval system for region of interest	83
3.22 LSS descriptor	86
3.23 Retrieval performance of our proposed methods and the MSU method using CMC in the ROI dataset with background image	88
3.24 Retrieval performance of our proposed methods and MSU method using CMC in ROI dataset without background image	90
3.25 Retrieval performance of our proposed methods and MSU method using precision and recall for the ROI dataset with background images	92
3.26 Retrieval performance of our proposed methods and MSU Method using precision and recall for the ROI dataset without background images	93
4.1 System overview	99

Figure	Page
4.2 Example of Center-Aligned Spatial Pyramid Generation: Blue color represents the original spatial pyramid grid and red color represents our CASP grid	101
4.3 Our Modified Center-Aligned Spatial Pyramid (MCASP) structure . . .	103
5.1 Our proposed shape retrieval system	112
5.2 Sample shape images in our database	118
6.1 Overview of the Tattoo Image Analysis System	123
6.2 An Example of a Login	124
6.3 User Options: (a) is the main screen and (b) is the secondary screen . . .	125
6.4 An Example of Image Browsing	126
6.5 Four Different Options in Browse Database	127
6.6 Examples of Browsing Database By Radius	127
6.7 Examples of Browsing Database By Date	128
6.8 Examples of Browsing Database By Gang Name	129
6.9 An Example of Image Capture	129
6.10 Examples of Sending Images to Server	130
6.11 Examples of Finding Similar Images	131
6.12 An Example of the Setting Option	132
6.13 An Example of Login	132
6.14 Main Menu in Web Interface	133
6.15 An Example of Browse Database by Date	135
6.16 An Example of Browse Database by Gang Name	136
6.17 An Example of Browse Database by First Responder ID	137
6.18 An Example of Finding Similar Images	138

ABSTRACT

Kim, Joonsoo. Ph.D., Purdue University, August 2017. Context Based Tattoo Image Analysis with Applications in Public Safety. Major Professor: Edward J. Delp.

Law enforcement is interested in exploiting tattoos as an information source to identify, track and prevent gang-related activities. In this thesis we examine several aspects of tattoo image analysis and how to extract useful information from tattoo images. There are problems with existing tattoo image systems. For example, many existing tattoo retrieval systems do not use local and global image descriptors robust to deformations caused by "manually constructed" tattoos on human skin. One other issue is that most tattoo images are manually cropped to remove background clutter from the image before analysis.

In this thesis we examined various aspects of a tattoo image retrieval and classification, in particular we investigated segmentation, classification, image matching and retrieval. A tattoo region is segmented using graph-cut tattoo segmentation based on image edges, a skin color model and a visual saliency map. We generate local and global image descriptors for the segmented image based on multiple polar histograms to introduce robustness against various deformations. The multiple polar histograms are combined with SIFT descriptors in the local image descriptor and 2D DFT in the global image descriptor. We then search our tattoo image database and retrieve images similar to the segmented image using an image matching technique based on both descriptors. To improve the image retrieval accuracy, not only we find the pairwise image similarity between an input image and a database image but we also incorporate all the image similarities between the database image and other database images using inductive matching.

For the tattoo image classification, sparse codes based on dense SIFT descriptors are generated. The sparse codes are then combined with a spatial pyramid feature pooling to incorporate the spatial distribution of the sparse codes. A spatial pyramid alignment method is additionally used to improve the image classification accuracy. These methods are evaluated on datasets that were collected from the Indiana State Police, eviltattoo.com, and the NIST tattoo challenge dataset.

1. INTRODUCTION

1.1 Tattoo Segmentation

Tattoos are very important in monitoring criminal gang activities and gang members. A large percentage of gang members use tattoos to “display” their gang affiliation and their identities in criminal activities. Therefore, law enforcement has been interested in exploiting tattoos as an information source to investigate gang-related activities as well as for gang member identification.

Many tattoo image retrieval systems, that retrieve similar tattoo images from a tattoo image database, have been proposed [1–10]. For the accurate image retrieval, most of them use the tattoo segmentation to crop the tattoo regions only.

The importance of the tattoo segmentation is also addressed in the Tattoo Recognition Technology - Challenge (Tatt-C) [11]. NIST (National Institute of Standards and Technology) conducted the Tatt-C in early 2015 for developing tattoo image recognition methods [12]. They provided the Tatt-C database that consists of five datasets focused on five primary use cases: tattoo identification, region of interest, mixed media, tattoo similarity, and tattoo detection [12, 13]. After reviewing the tattoo recognition methods submitted in the Tatt-C, they reported that the accurate tattoo segmentation (or localization) is important to improve the tattoo recognition accuracy [11].

Many existing tattoo image retrieval methods manually crop the tattoo image to remove varying background from an image because there are lots of challenges in the tattoo segmentation. For example, the skin detection, which is the initial step in tattoo segmentation, sometimes fails when the background has color similar to skin tones, there is an illumination change, or different skin tones should be detected.

Also, when there is background clutter, many existing image segmentation methods could not find the tattoo region correctly.

Many tattoo image segmentation methods have been proposed [1, 2, 5, 7–9, 14–17]. In [7–9] tattoo segmentation is used to extract a tattoo shape as well as to remove varying background from an image. In [7, 9] a Sobel operator and morphological operators are used to extract low level features such as color, texture and shape for tattoo image retrieval. Instead of using a morphological operator, which is not robust to weak edges, active contour based segmentation [18] is used in [9]. However, the segmentation methods in [7–9] use pre-cropped images and it is assumed that the background is mostly skin and homogeneous.

In [1, 2, 5, 14, 15] efficient tattoo segmentation methods are introduced for non pre-cropped images. The use of the HSV color space to find skin regions is investigated to segment tattoos in [1, 15]. In [5] tattoo segmentation is done using skin detection followed by a figure-ground segmentation. In [2] a visual saliency map model is used along with Grabcut [19] and QCC (Quasi Connected Components) [20] for tattoo segmentation. In [14] the tattoo region detection is used to de-identify the tattoo region for privacy protection. This method detects the skin regions using the skin color model roughly and uses the SIFT matching [21] to find the tattoo region more precisely.

Since the performance of a deep neural network has been proved in the image segmentation as well [22–25], the deep neural network learning based tattoo localization methods have been proposed [16, 17]. To train a deep neural network for a segmentation, the pair of an image and a ground truth segmentation map is necessary. Once the neural network is trained, a test image will be presented into the network, and it will generate the segmentation map that labels each pixel in the test image. Like the neural network for image recognition, CNN (convolutional neural network) architecture [26] shown in Figure 1.1 is also used in the neural networks for segmentation. However, a fully connected layer, which is used in CNN, is not used in this network. Instead of the fully connected layer, convolutional layer is used because the

neural network for image segmentation should generate a segmentation map while the network for image recognition generates an object class vector.

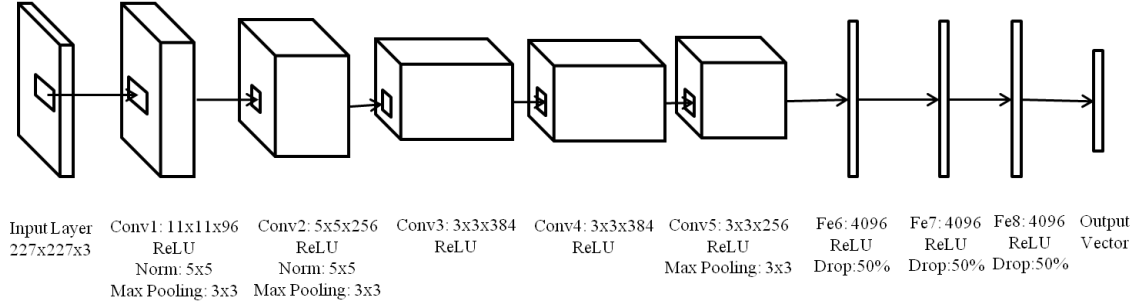


Fig. 1.1.: The Convolutional neural network (CNN) architecture

In [16] CNN [26] is trained using the tattoo image patches. Once the CNN is trained, all the image patches in a test image are presented into the CNN and they are classified as the tattoo patches or the non-tattoo patches. In [17] the faster R-CNN [24, 27] is customized to localize tattoo regions and classify the tattoo object together. Given the locations and the class labels of the tattoo objects in an image, the faster region proposal convolutional neural network (F-RCNN) is trained first. For test, the location of a tattoo object and the corresponding tattoo class label are estimated together. This method achieves high localization accuracy, but it requires lots of ground truth images that include the tattoo locations and the tattoo class labels. Also, the localization accuracy is highly dependent on the region proposal methods used in the network.

1.2 Tattoo Image Retrieval Based on Image Matching

A large percentage of criminal gang members use tattoos to show gang affiliation and to draw attention to events in their lives related to criminal activity. For this reason law enforcement is interested in exploiting tattoos as an information source to identify, track and prevent gang-related crimes. One application scenario is the use of tattoo information at the time a suspect is arrested. One would like to be

able to retrieve similar tattoos from a tattoo image database and determine who that individual might be associated with. Another application is to recognize gang symbols or other information depicted in the tattoo. Since a tattoo is “manually constructed” on a surface (skin) that has irregular properties, a tattoo image contains deformations that can affect the image matching process. Also, tattoos have weak edge structures, adding more difficulties for image analysis and matching. The combination of ink color, skin color, skin textures (e.g. scars) and hair add even more challenges.

There exist many methods focused on tattoo image retrieval [4, 6–9, 15, 17, 28–34]. In [9] low level features such as color, texture and shape are used for a content-based tattoo image retrieval system. Instead of using a morphological operator, which is not robust to weak edges, active contour based segmentation [18] is used to extract a tattoo from an image in [8]. In [7] a new rank-based distance metric learning method is described for a tattoo image retrieval system based on low level attributes. In [28] the same authors insists that a tattoo image retrieval system should be based on the concept of “visually similarity,” which can narrow the “semantic gap” [35].

SIFT (Scale-invariant feature transform) [21] has been widely used to combine with a Bag-Of-Words (BOW) [36] model in [3, 37, 38]. In [37] the computational complexity of SIFT feature clustering is examined when the features are quantized in the BOW model. In [38] the quantization error caused by feature clustering in the BOW model is further studied using multiple BOW models and combined using weighted averages. The quantization error problem in the BOW model is also addressed in [3] using Hamming distance and geometry consistency to improve retrieval accuracy.

In [4, 6, 29–32] various matching based tattoo image retrieval systems are described. Matching a tattoo sketch used in the identification of a suspect with a real tattoo image is discussed in [29, 31]. In [6] geometric constraints of SIFT are used to improve image matching accuracy. A robust similarity metric is described for SIFT based image matching in [4]. In [32] an accurate feature extraction method based on higher

order scale space is addressed. In [30] they show that the image registration between two tattoo images can improve the image recognition accuracy.

Since the deep neural network architecture has shown its success in an image retrieval [34, 39, 40], many deep neural network based methods have been proposed to recognize and retrieve a tattoo object [15, 17, 33, 34]. One of the most famous networks for an image retrieval is Siamese network [39]. This network is composed of two independent CNNs. Each CNN accepts an image as an input. Note that all the images should have the same size. The output of each CNN is then compared with each other using a distance metric. By minimizing the difference of the outputs of two CNNs based on the distance metric, the Siamese network is learnt.

In [33], CNN (convolutional neural networks) [26] is customized to determine if an image includes a tattoo. This network consists of five convolutional layers and three fully connected layers. The experiment shows that a CNN based tattoo classification outperforms all the methods reported in NIST tattoo challenge [11]. The correlation neural network is used to classify the tattoo object after the tattoo localization in [15]. This network consists of four layers: the first layer is an input layer, the second layer is a fully connected layer, the third layer is a fully connected layer but it has less number of neurons than the second layer, and the last layer is an output layer whose a neuron represents the class of similar tattoos. In [34], they use Siamese network [39] for matching two tattoo images and convolutional neural network (CNN) for classifying a tattoo image. This Siamese network consists of two CNNs. Each of an input image and a reference image is presented into each CNN. The output of each CNN is then compared with each other using the triplet loss function [41]. By minimizing the loss function, the Siamese network is learnt. The experiment shows that the Siamese network based image matching method outperforms all the methods reported in [11] for the tattoo similarity dataset [13]. In [17] the faster R-CNN [24, 27] is used to localize tattoo regions and classify the tattoos at the same time. The region proposal [42–45] that can include tattoo objects is extracted first. The extracted region proposal is presented into CNN as an input image. The object class vector is

generated as the output of the R-CNN, and the class with the maximum element of the vector is chosen as the class of the object in the region proposal.

1.3 Tattoo Image Classification Based On Sparse Coding

For image classification, the bag of visual words (BOW) [46] model has been widely used. Among several BOW models, the sparse coding method has been more popular than any other methods because the sparse coding based image representation is more robust to the appearance and shape variations of the image objects that belong to the same class. Since the tattoo images have lots of the appearance and shape variations, we use the sparse coding technique to represent the tattoo image.

To represent an image using the BOW model, the local image feature such as SIFT is extracted first. Then, each local feature is coded using a set of predefined codewords. The set of codewords is referred to as the codebook. Coding features using the codebook is analogous to representing vectors using a set of basis vectors. The codebook is usually constructed using a set of local image features randomly sampled from the training images. The feature coding vector is the output of the coding process. It is comprised of a set of coefficients where each coefficient is the contribution of a particular codeword in representing the feature. Vector quantization (VQ) [47,48] simplifies the coding process by assuming that a feature can be represented by a single codeword. Therefore, all the elements of the coding vector are zeros except for a single element corresponding to the codeword closet to this feature. However, vector quantization is not adequate to represent the variation of features. This causes degradation in the performance of image representation and classification. Sparse coding [49–59] has been utilized to address this problem. In sparse coding each local image feature is represented by a combination of a small number of codewords. The final image representation is based on the coding vectors of all the local image features. To combine multiple coding vectors into a single vector, average or max pooling [60] is utilized. In average pooling, the final image representation vector is computed by

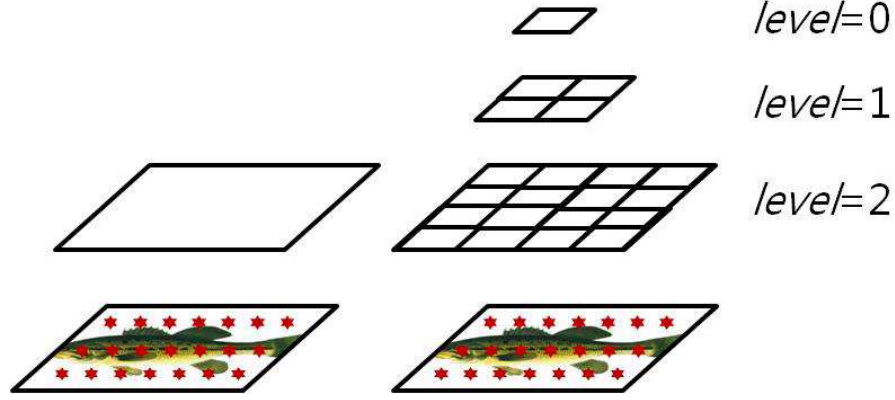


Fig. 1.2.: Example of the BOW model with the spatial pyramid

averaging all the coding vectors, whereas max pooling uses the maximum value of each element among all the coding vectors separately. Figure 1.2 shows an example where dense SIFT local features (represented by red stars) are extracted from an image. The basic BOW model combines the features coding vectors (left side in Figure 1.2) without considering the spatial distribution (layout) of the local image features. This means that the spatial locations of the local images features are not used in the BOW model. This drawback of the BOW model limits the descriptive power of the final image representation.

To address this problem, spatial pyramid feature pooling (SPP) [47,49,52–58] has been proposed and incorporated in most feature coding methods. To construct a spatial pyramid, an image is partitioned into $2^l \times 2^l$ subregions at different l^{th} levels ($l=0,1,2$), as shown in Figure 1.2 (right).

The first level consists of 16 subregions, whereas the second level contains 4 subregions and the third one is a single region. Instead of using max or average pooling on the entire image, SPP is done on each subregion. The final image representation of SPP is the concatenation of all the subregions representation vectors. Existing methods which are based on SPP assume that the center of an object is aligned with the center of the image. Therefore, the center of the image is used as the center of the spatial pyramid. However, the center of most images are not aligned with the

center of objects correctly. This misalignment propagates in feature pooling results in several subregions at multiple pyramid levels.

1.4 Contributions Of The Thesis

In this thesis we first investigate a tattoo segmentation method that removes background clutter from an image. We then introduce our tattoo image retrieval system based on the proposed local and global image descriptors. We also improve our previous image descriptor and image matching method for more accurate image retrieval. We describe our submissions to the Tatt-C Tattoo Identification (TID) and the Tatt-C Region of Interest (ROI). We propose our spatial pyramid alignment technique for sparse coding based object classification. This technique is used on the tattoo image dataset as well as public object recognition image datasets. Last, we introduce a shape descriptor known as Self Similar Affine Invariant (SSAI) descriptor for shape retrieval. The main contributions of this thesis are listed as follows:

- Efficient Graph-Cut Tattoo Segmentation

We propose a graph-cut tattoo segmentation method based on image edges, a skin color model, and a visual saliency map to find skin pixels around tattoo regions. The post processing, that detects the skin pixels around a tattoo only from the graph-cut segmentation results, is then used. The method was evaluated on datasets that were collected from the Indiana State Police, eviltattoo.com [61], and NIST tattoo challenge dataset [13]. Experimental evaluation demonstrates that our segmentation method can detect and segment tattoo regions correctly even when a tattoo image includes background clutter.

- Efficient Graph-Cut Tattoo Segmentation with Body Boundary Removal

We propose body boundary removal (BBR) method to improve our previous segmentation method. The previous method makes errors when tattoo regions are very close to the boundaries of human body. Thus, our BBR method fixes

the errors by removing the skin pixels on the boundaries of human body. Experimental results demonstrate that the segmentation errors of our previous method are reduced using BBR.

- Tattoo Image Matching and Retrieval Based on Local Image Descriptor (MHLC Descriptor) and Global Image Descriptor Robust to Image Deformations

We create new local and global shape descriptors robust to scale, translation, rotation, and shape distortions for tattoo image retrieval. By using the scale invariant feature transform (SIFT) with local shape context based on multiple different sized-bin polar histograms (MH), more accurate image matching can be obtained. A global shape descriptor based on MH and a 2D Fourier Transform is also used for robustness of translation, scale, rotation and shape distortions. We also describe robust similarity for local descriptors and a weighted matching method based on local and global descriptors. Experimental results show that our method outperforms several existing methods.

- Tattoo Image Matching and Retrieval Based on Modified MHLC Descriptor (DMHLC Descriptor)

We introduce the improved MHLC descriptor, called as DMHLC descriptor. Instead of using the spatial distribution of the SIFT features, the DMHLC descriptor uses the spatial distribution of the densely sampled features on the tattoo object to generate the multiple polar histograms. The multiple polar histograms are combined with the SIFT descriptor to generate DMHLC descriptor. Our experimental results show that our DMHLC descriptor improves the image retrieval accuracy much more than our MHLC descriptor.

- Modified Inductive Matching

We introduce our modified inductive matching to improve the image retrieval accuracy. By considering all the similarities between all the images in the database, the image retrieval accuracy is improved. The modified inductive

matching retrieves the most dissimilar M_2 database images respect to an input image first. Then, the mean of the image similarities between the M_2 database images and one database image is considered to compute the final image similarity between the input image and the database image. Our experimental results show that the modified inductive matching improves the image retrieval accuracy more than the pairwise image similarity based on the image matching of two images.

- The Our Submissions to NIST Tattoo Recognition Technology Challenge

For tattoo image retrieval on NIST Tattoo Identification (TID) dataset, the tattoo image retrieval system based on the MHLG descriptor is introduced. Experimental results show that our method outperforms the method of [4]. Our method also outperforms five different methods reported in the NIST challenge [11]. For tattoo image retrieval on NIST Region of Interest (ROI) dataset, we also create another image descriptor based on local self similarity (LSS) [62] and SIFT. We also propose a weighted distance similarity metric to retrieve the most similar images from the test dataset. Experimental results demonstrate that our method outperforms the method of [4]. Our method also outperforms four different methods reported in the NIST challenge [11].

- Spatial Pyramid Alignment For Sparse Coding Based Object Classification

We propose a simple but efficient spatial pyramid alignment method that can be combined with the existing sparse coding methods. By using max pooled features, we estimate an object center and align the spatial pyramid accordingly. We also propose an image representation descriptor robust to misalignment and object deformations using max pooling on multiple image descriptors generated by shifting the pyramid center in a pre-defined margin. We test the modified center-aligned spatial pyramid with the sparse coding method on the tattoo image dataset as well as public object recognition image datasets. Our experimental results show that our proposed spatial pyramid with the sparse coding

improve the image object classification accuracy more than the original spatial pyramid with the same sparse coding.

- Shape Matching Using A Self Similar Affine Invariant Descriptor

We introduce a shape descriptor known as Self Similar Affine Invariant (SSAI) descriptor for shape retrieval. The SSAI descriptor is based on the property that two sets of points are transformed by an affine transform, then subsets of each set of points are also related by the same affine transformation. Also, the SSAI descriptor is insensitive to local shape distortions. We use multiple SSAI descriptors based on different sets of neighbor points to improve shape recognition accuracy. We also describe an efficient image matching method for the multiple SSAI descriptors. Experimental results show that our approach achieves very good performance on two publicly available shape datasets.

1.5 Publications Resulting From Our Work

Journal Papers

1. **J. Kim** and E. J. Delp, “Tattoo Image Retrieval Based On Robust Tattoo Image Matching”, *To be submitted to the IEEE Transactions on Information Forensics and Security*.
2. **J. Kim**, K. Tahboub, and E. J. Delp, “Shape Matching and Retrieval Using a Self Similar Affine Invariant Descriptor”, *To be submitted to the IEEE Signal Processing Letters*.

Conference Papers

1. **J. Kim**, K. Tahboub, and E. J. Delp, “Spatial Pyramid Alignment For Sparse Coding Based Object Classification”, to appear *Proceedings of the IEEE International Conference on Image Processing*, Beijing, China.
2. **J. Kim**, H. Li, J. Yue, and E. J. Delp, “Shape Matching Using a Self Similar Affine Invariant Descriptor”, *Proceedings of the IEEE International Conference on Image Processing*, pp. 2470-2474, September, 2016, Phoenix, AZ.
3. **J. Kim**, H. Li, J. Yue, J. Ribera, L. Huffman, and E. J. Delp, “Automatic and Manual Tattoo Localization”, *Proceedings of the IEEE International Conference on Technologies for Homeland Security*, pp. 1-6, May 2016, Waltham, MA.
4. **J. Kim**, H. Li, J. Yue, and E. J. Delp, “Tattoo Image Retrieval for Region of Interest”, *Proceedings of the IEEE International Conference on Technologies for Homeland Security*, pp. 1-6, May 2016, Waltham, MA.
5. **J. Kim**, A. Parra, J. Yue, H. Li, and E. J. Delp, “Robust Local and Global Shape Context for Tattoo Image Matching”, *Proceedings of the IEEE International Conference on Image Processing*, pp. 2194-2198, October 2015, Quebec, Canada.

6. **J. Kim**, A. Parra, H. Li, and E. J. Delp, “Efficient Graph-Cut Tattoo Segmentation”, *Proceedings of the SPIE/IS&T Conference on Visual Information Processing and Communication VI*, pp. 94100H-1-8, February 2015, San Francisco, CA.
7. A. Parra , B. Zhao, **J. Kim**, and E. J. Delp, “Recognition, Segmentation and Retrieval of Gang Graffiti Images on a Mobile Device”, *Proceedings of the IEEE International Conference on Technologies for Homeland Security*, pp. 178-183, November 2013, Waltham, MA.

2. TATTOO SEGMENTATION

2.1 Review of Existing Methods

In image retrieval and recognition applications, image segmentation is a significant preprocessing step to improve the image retrieval and recognition accuracy [24,27,63–67]. As the result of image segmentation, object regions are segmented (or localized) from an image and the shape of the object is obtained. The segmented image is then presented into the image retrieval or recognition system to find the images with similar object or to classify the segmented image. The main purpose of the segmentation is to focus on the objects only by removing background clutters.

For the same reasons, the tattoo segmentation (or localization) is a very important step in the tattoo image retrieval or the tattoo image classification. In [11,68], they show that the accuracy of tattoo image retrieval or classification can be dropped significantly when the tattoo segmentation is not used.

There have been many image segmentation methods such as graph based segmentations [19,69,70], active contour based segmentations [18,71,72], and deep learning based segmentations [22–24].

An active contour (snake) based segmentation [18,71,72] evolves a closed contour from some initial position toward the actual boundary of an object. The initial position of the contour is generally specified by a human or is roughly determined by an object detection method. Once the initial contour is determined, the evolution of the contour is done by minimizing an energy function. The energy function generally consists of two energy terms: an internal force energy and an external force energy. The internal force energy evolves the contour more smooth while the external force energy makes the contour to be near the boundary of an object.

In graph based segmentation [19, 69, 70, 73], an image pixel (or the group of the image pixels close to each other) is considered as a node of a graph. The relationships between the image pixel and the neighbors of the pixel are represented as the edges between nodes in the graph. Once the graph is constructed in an image, the segmentation problem is formulated as a graph partition problem. Many graph optimization methods [70, 73, 74] can be then used to solve the segmentation problem. One of the most famous graph based segmentation methods is a graph-cut segmentation [69, 73]. In the graph-cut segmentation, a segmentation problem is formulated as 2-class labeling (foreground or background) problem. For example, the pixel, i , is labeled by minimizing the following Gibbs energy:

$$E(x) = \sum_{i \in I} D(x_i) + \lambda \sum_{i \in I, j \in N_i} V(x_i, x_j), \quad (2.1)$$

where i is a pixel, I is an image, N_i are neighbors of pixel i , $D(x_i)$ is the data term, $x_i \in \{0(\text{background}), 1(\text{foreground})\}$ is a label for i , and $V(x_i, x_j)$ is the smoothness term. Depending on applications, the different data term and the smoothness term can be defined to achieve the best performance. Once all the energy terms are defined, the Gibbs energy is minimized using the graph optimization such as min-cut/max flow in [74] and normalized cut [70].

In the deep learning based segmentations [22–25], the pair of an image and a ground truth segmentation map is necessary to train the deep neural network. Once the neural network is trained, a test image will be presented into the network, and it will generate the segmentation map that labels each pixel in the test image. Like the neural network for image recognition, CNN architecture [26] is also used in the neural networks for segmentation. However, a fully connected layer, which is used in CNN, is not used in this network. Instead of the fully connected layer, convolutional layer is used because the neural network for image segmentation generates a segmentation map while the network for image recognition generates an object class vector.

One of the most famous neural network for image segmentation is SegNet [22]. As shown in Figure 2.1, SegNet consists of multiple convolutional layers such as convolu-

tional layers and deconvolutional layers. Convolutional layers (or called convolutional encoder) make the size of the input in each layer to be smaller using pooling while the deconvolution layers (or called convolutional decoder) make the size of the input in each layer to be bigger using upsampling. That is because the convolutional layer should approximate the input in each layer to be robust to image deformations while the deconvolutional layer should make the size of segmentation map the same as an input image.

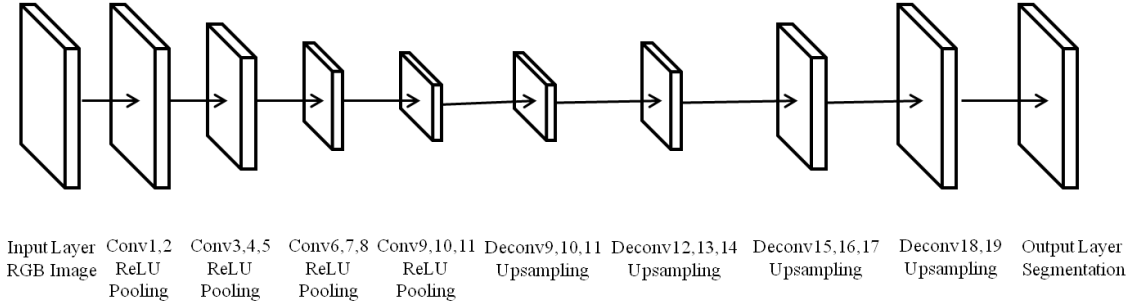


Fig. 2.1.: The SegNet architecture

These segmentation techniques have also been investigated in tattoo segmentation for content-based tattoo image retrieval. In [7,9] a Sobel operator and morphological operators are used to extract low level features such as color, texture and shape for tattoo image retrieval. The Sobel operator is used to compute the magnitudes and orientations of gradients at each pixel. The morphological closing and opening operations are then used on the regions where the magnitudes of gradients are large to find a tattoo region. The color, texture and shape features are then extracted on the detected tattoo region. Instead of using a morphological operator, which is not robust to weak edges, active contour based segmentation [18] is used in [8]. An area open-close filter [75] is used to remove small scale objects and noise first. The contour of a tattoo region is detected using Vector filed convolution (VFC) contour based segmentation [18]. A skin detection method is then used to extract the tattoo region only. The same low level features (color, texture, and shape) extracted from the region are used to generate an image descriptor for a tattoo. However, the segmentation

methods in [7–9] use pre-cropped images and it is assumed that the background is mostly skin and homogeneous. The methods are also sensitive to textures, so they often fail to separate the tattoo from complex backgrounds.

In [1, 2, 5, 14, 15], efficient tattoo segmentation methods are introduced for non pre-cropped images. The use of the HSV color space to find skin regions is investigated to segment tattoos in [1, 15]. Once the skin regions are detected, the negative transformation is used on the regions to segment a tattoo region only. However, the fixed color range defined to detect skin pixels is restricted to Asian people, and it can be affected by illumination, background, and camera characteristics. In [5] tattoo segmentation is done using skin detection followed by a figure-ground segmentation. Based on the assumption that the center region of a tattoo image contains skin, the regions including both skin and tattoo are first detected by merging the regions connected with the center region. Then, a figure-ground segmentation using k-means ($k=2$) clustering [76] is used in the RGB color space to distinguish a tattoo region from a skin region. However, the assumption is not true in general and k-means ($k=2$) clustering in the color space cannot distinguish skin and tattoo when a tattoo has various colors. In [2] a visual saliency map model is used along with Grabcut [19] and QCC (Quasi Connected Components) [20] for tattoo segmentation. First, the region around a tattoo is detected based on a visual saliency model. Then, Grabcut segmentation [19] is used in the region to localize a tattoo region in more detail. In parallel QCC is used in the detected edge map to find the connected regions that have lots of textures. The final tattoo region is then segmented using the results of Grabcut and QCC. However, the accuracy of this segmentation method highly depends on the accuracy of a visual saliency map. In [14], a tattoo region detection method is introduced to de-identify the tattoo region for privacy protection. This method detects skin regions first using the skin color model and the geometric constraint. The holes and cutout regions close to the detected skin regions are combined to formulate the region of interest (ROI). From the ROI SIFT features [21] are extracted, and SIFT feature matching against the tattoo images in the database is done to find the tattoo

region only. However, the SIFT matching based tattoo region detection could be incorrect when there are not similar tattoo images in the database.

The deep neural network learning based tattoo localization methods have been proposed [16, 17]. In [16], CNN (convolutional neural networks) [26] is trained using the tattoo image patches. Once the CNN is trained, all the image patches in a test image are presented into the CNN and they are classified as the tattoo patches or the non-tattoo patches. The regions including the detected tattoo patches are roughly considered as initial tattoo regions. The morphological operations is then used in the regions to find tattoo blobs. The final tattoo regions are detected to find a closed contour on the tattoo blobs. In [17] the faster R-CNN [24, 27] is customized to localize tattoo regions and classify the tattoo at the same time. Given the locations and the class labels of the tattoo objects in an image, the faster region proposal convolutional neural network (F-RCNN) is trained first. For test, the location of a tattoo object and the corresponding tattoo class label are estimated together. This method achieves high localization accuracy, but it requires lots of ground truth images that include the tattoo locations and the tattoo class labels. Also, the localization accuracy is highly dependent on the region proposal methods used in the network.

2.2 Efficient Graph-Cut Tattoo Segmentation

We define the tattoo segmentation problem as finding skin pixels around a tattoo assuming a tattoo is surrounded by skin. Therefore, we do not need to segment all pixels in the image. The regions near image edges are the only ones considered for segmentation (there are always edges between tattoo and skin). We call these regions “possible segmentation regions”. We detect skin pixels in the regions near image edges using a probabilistic skin color model based on a Gaussian Mixture Model.

If there are regions that have color similar to skin in the background the skin color model fails to detect the skin around the tattoo. We additionally use a visual saliency map to focus on skin near tattoo regions. Our problem can be then considered as

a 2-class labeling problem to classify each pixel around edges as skin or non-skin. A variation of graph-cut segmentation [69] using a skin color model and a visual saliency map is used in our proposed system. After the segmentation we check which set of skin pixels are connected with each other that forms a closed contour including a tattoo. The region surrounded by the closed contour is considered as a tattoo region. A block diagram of our proposed tattoo image segmentation system is shown in Figure 2.2.

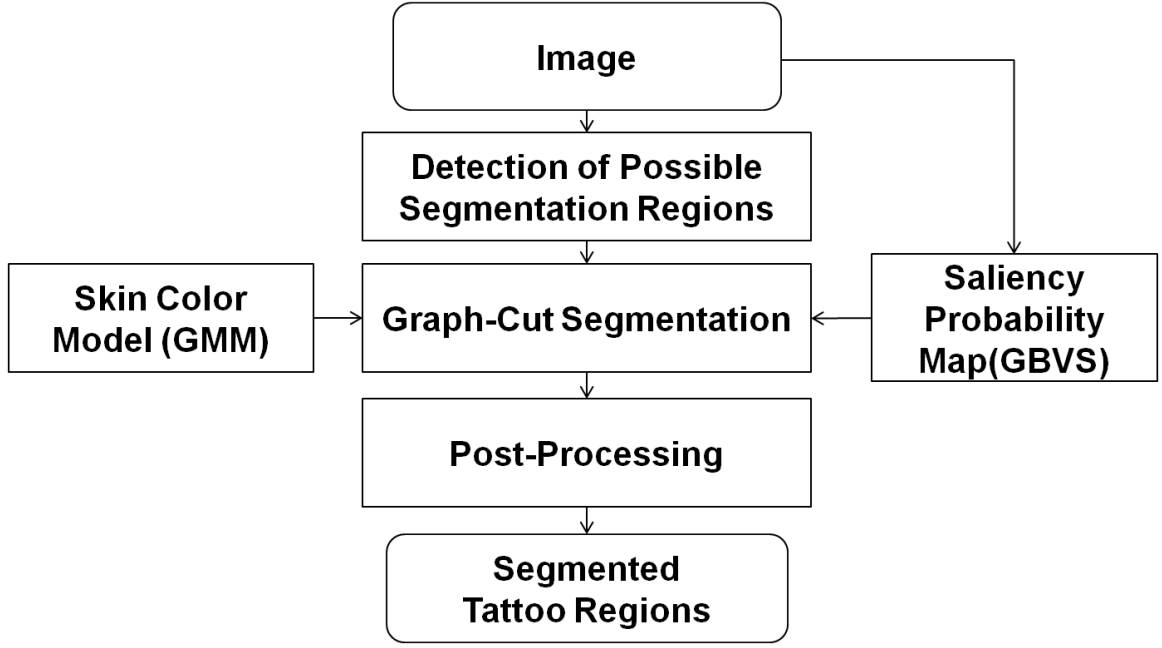


Fig. 2.2.: Our proposed tattoo image segmentation system

2.2.1 Detection of Possible Segmentation Regions

In this section we describe how the possible segmentation regions are detected. Since we only focus on regions near edges, edge detection is done first. The Canny edge detector [77] is used to find edges for each RGB color channel separately. The edge regions from each channel are then combined. Morphological dilation is then used to the combined the edge regions to find the regions near edges. These regions

are considered as “possible segmentation regions”. In Figure 2.3(b) white regions are possible segmentation regions. There are two advantages to limiting the regions for segmentation. First, the segmentation errors outside of the possible segmentation regions can be avoided. Second, the segmentation execution time can be reduced.

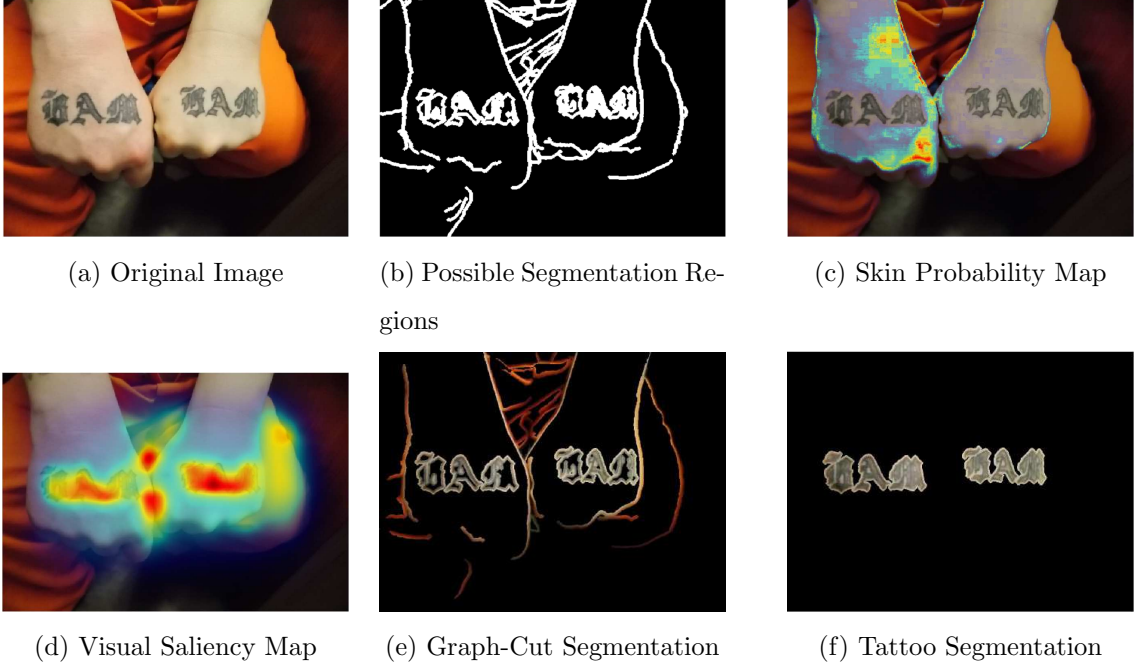


Fig. 2.3.: Overall segmentation process

2.2.2 Graph-Cut Segmentation

Once the regions for segmentation are determined, we find skin pixels by using a 2-class labeling approach. Graph-cut segmentation [69] is described by the Gibbs energy:

$$E(x) = \sum_{i \in I} D(x_i) + \lambda \sum_{i \in I, j \in N_i} V(x_i, x_j), \quad (2.2)$$

where i is a pixel, I is an image, N_i is neighborhood pixels of pixel i , $x_i \in \{0(background), 1(foreground)\}$, $D(x_i)$ is the data term, and $V(x_i, x_j)$ is the smooth-

ness term. To construct the graph for this energy, each pixel in an image is considered as a graph node and two nodes for foreground and background are added in the graph. Then, the data term is obtained by connecting each pixel to both the foreground and background nodes with non-negative edge weights represented by $D(x_i = 1)$ and $D(x_i = 0)$. The smoothness term is obtained by connecting each pairwise combination of neighboring pixels (i, j) with a non-negative edge weight represented by $V(x_i, x_j)$. We used the graph-cut segmentation approach described in [69] by modifying the data term, $D(x_i)$ for tattoo segmentation. For skin detection, we define the data term as:

$$D(x_i) = w_1 D_1(x_i) + w_2 D_2(x_i), \quad (2.3)$$

where $D_1(x_i)$ is the energy for the skin or non-skin color model, $D_2(x_i)$ is the energy for the visual saliency map, $x_i = 1$ means skin, $x_i = 0$ means non-skin, and w_1 and w_2 are weights for $D_1(x_i)$ and $D_2(x_i)$. Then, each energy term is defined as:

$$\begin{aligned} D_1(x_i = 1) &= -\log(p_1(x_i = 1)), \quad D_2(x_i = 1) = -\log(p_2(x_i = 1)) \text{ for } i \in PS \\ D_1(x_i = 0) &= \infty, \quad D_2(x_i = 0) = \infty \text{ for } i \in I - PS \end{aligned} \quad (2.4)$$

where PS are the possible segmentation regions.

The skin models illustrated in [78–85] described the fixed skin color ranges in different color space to detect skin. However, the skin detection based on the fixed color ranges cannot model diverse skin colors. Since the each mixture component of the GMM can model the different skin tone, we use a Gaussian Mixture Model (GMM) for $p_1(x_i = 1)$ to train the skin color model on diverse skin tones. It is defined as:

$$p(x_i = 1) = \sum_{j=1}^M \pi_j * g(C_i | u_j, \Sigma_j) \quad (2.5)$$

where C_i is the YCbCr color of i^{th} pixel, π_j is the j^{th} mixture weight, M is the number of mixture components, and $g(C_i|u_j, \Sigma_j)$, $j=1, \dots, M$, are the component Gaussian densities. The component Gaussian density, $g(C_i|u_j, \Sigma_j)$ is defined as:

$$g(C_i|u_j, \Sigma_j) = \frac{1}{(2\pi)^{3/2}|\Sigma_j|^{1/2}} \exp\left\{-\frac{1}{2}(C_i - u_j)^T \Sigma_j^{-1} (C_i - u_j)\right\} \quad (2.6)$$

To estimate the skin color model parameters, u_j , Σ_j , and π_j we use EM (Expectation-Maximization) [86] on a skin image dataset obtained from [87]. For $p_2(x_i = 1)$, we use the visual saliency map from the GBVS (Graph-based visual saliency) [88]. To generate the saliency map from the GBVS, feature vectors are extracted using the method of [89] first. Low-level visual features (color, contrast, and an orientation) are used as the feature vectors with a Gaussian pyramid in [89]. Then, it generates each saliency map (activation map) using each feature vector. Lastly, it normalizes each saliency map and combines all saliency maps together using graph-based approach. Since the GBVS saliency map does not have the same resolution as the image we upsample the map using bicubic interpolation. In [90] GMM is also used to train non-skin color as well as skin color on the skin images. However, We do not use the GMM to train the non-skin color model because we cannot estimate all non-skin colors. Instead, we use an adaptive threshold value for $D_1(x_i = 0)$. Similarly, another adaptive threshold value is used for $D_2(x_i = 0)$:

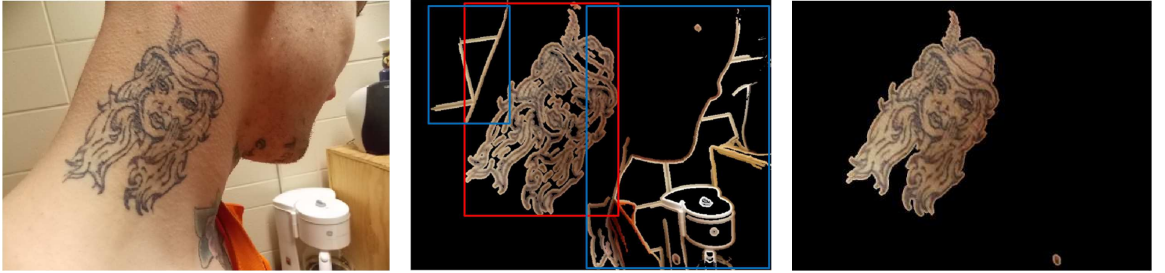
$$\begin{aligned} D_1(x_i = 0) &= -\log(wt_1 * \frac{1}{n_I} \sum_{i \in I} p_1(x_i = 1)) \text{ for } i \in PS \\ D_2(x_i = 0) &= -\log(wt_2 * \frac{1}{n_I} \sum_{i \in I} p_2(x_i = 1)) \text{ for } i \in PS \\ D_1(x_i = 0) &= 0, \quad D_2(x_i = 0) = 0 \text{ for } i \in I - PS \end{aligned} \quad (2.7)$$

where I is an image, n_I is the number of pixels in I , and wt_1 and wt_2 are weights for thresholds. Since we use adaptive thresholds instead of fixed thresholds for $D_1(x_i = 0)$ and $D_2(x_i = 0)$ the labeling errors can be reduced when the skin colors in an image

do not fit our skin color model or a GBVS does not fit the image. The smoothness term, $V(x_i, x_j)$ is defined as:

$$V(x_i, x_j) = |x_i - x_j| * f(C_{ij}) \quad (2.8)$$

where $f(\xi) = \frac{1}{1+\xi}$, and $C_{ij} = ||C_i - C_j||^2$ is the L2-Norm of the YCbCr color difference of two pixels i and j . This smoothness term is similar to the smoothness term in [69], but we use YCbCr color space for C_i while the RGB color space was used for C_i in [69]. The overall Gibbs energy, $E(x)$ is then minimized by min-cut/max flow in [74]. The min-cut/max flow method consists of three stage: growth stage, augmentation stage, and adoption stage. In the growth stage it expands the search trees S and T until they touch each other with generating an $s - t$ path(s : a node for foreground, t : a node for background). In augmentation stage the path found in the growth stage is augmented. The adoption stage restores single-tree structure of sets S and T with roots in s and t.



(a) Original Image

(b) Graph-Cut Segmentation

(c) Final Tattoo Segmentation

Fig. 2.4.: Examples of false contours: In Figure 2.4(b), the red rectangular box is the minimum bounding box for a tattoo region, and the blue rectangular boxes are the minimum bounding boxes which include false contours. Using additional constraint, Equation (2.9), a tattoo region is only segmented as depicted in Figure 2.4(c)

2.2.3 Post-Processing

From the graph-cut segmentation the skin pixels near edges are detected as shown in Figure 2.3(e). Our method also detects skin pixels near the boundaries of human body as well as the skin pixels near tattoos. To extract the skin pixels only near tattoos we check which set of skin pixels connected with each other forms a closed contour with a hole inside because the tattoo pixels surrounded by skin pixels will be labeled as non-skin and it will formulate holes. We can find the skin pixels near tattoos by finding the connected components of segmented pixels which have holes inside. As shown in Figure 2.4(b), however, skin pixels near the boundaries of the body might formulate closed contours with small holes inside because of segmentation errors. We will call these closed contours false contours. To distinguish a closed contour near tattoo with the false contours we additionally check the ratio of the area of minimum bounding rectangular box including the closed contour to the area surrounding all pixels connected with the closed contour. Note that if several contours are connected to each other, a contour including all the connected contours is only considered as the closed contour in this post processing.

$$\frac{n_c + n_h}{n_b} \geq t_f \quad (2.9)$$

where n_c is the number of pixels in a closed contour, n_h is the number of pixels for a hole inside the closed contour, n_b is the number of pixels in the minimum bounding rectangular box including the closed contour, and $t_f=0.35$ is the threshold to detect a false contour. If a closed contour satisfies Equation (2.9) the closed contour with inside regions is considered to be a tattoo region.

2.3 Efficient Graph-Cut Tattoo Segmentation With Body Boundary Removal

Our earlier segmentation method described in Section 2.2 makes errors when tattoo regions are very close to the boundaries of human body. That is because our

post processing cannot distinguish the skin pixels around a tattoo from the skin pixels near a body boundary when they meet each other. To address this problem, we propose body boundary removal. We observe that the regions near a body boundary mostly have simpler textures than the regions near the boundary of a tattoo. Our proposed body boundary removal (BBR) is based on a histogram of gradient orientation and will remove a pixel that has simple textures in a window region centered on the pixel. Figure 2.5 shows a block diagram of our tattoo segmentation system with BBR. The regions are detected using the Canny edge operator followed by a morphological dilation. Graph-cut segmentation based on a skin color model and a visual saliency map is used to detect skin pixels near tattoo regions. Skin pixels near the boundary of human body are removed from the detected skin pixels using BBR. A parameter of the BBR that determines the degree of the body boundary removal is iteratively chosen for the particular image. The steps, except for the BBR are the same as Section 2.2, so we describe the proposed BBR in more detail.

2.3.1 Body Boundary Removal (BBR)

A human body has mostly a smooth boundary and simple textures while most tattoos have complex boundaries and textures. To measure the degree of texture complexity a histogram of gradient orientation, weighted by gradient magnitude, is used as our texture descriptor. First, a moving window, W_i is centered on a segmented pixel i resulting from the graph-cut segmentation. A histogram of gradient orientation weighted by gradient magnitude, $h_i(m)$, is then computed for W_i . If the maximum value of $h_i(m)$ is large, we say that W_i has simple texture. However, a large maximum value of $h_i(m)$ can be split into two adjacent bins by quantization errors. To address this we compute the sum of the histogram for two adjacent bins which include the maximum of $h_i(m)$. The maximum bin index, m_{max} satisfying (2.10) is first computed.

$$m_{max} = \underset{m}{\operatorname{argmax}} h_i(m) \quad \text{for} \quad m = 1, 2, \dots, N_m \quad (2.10)$$

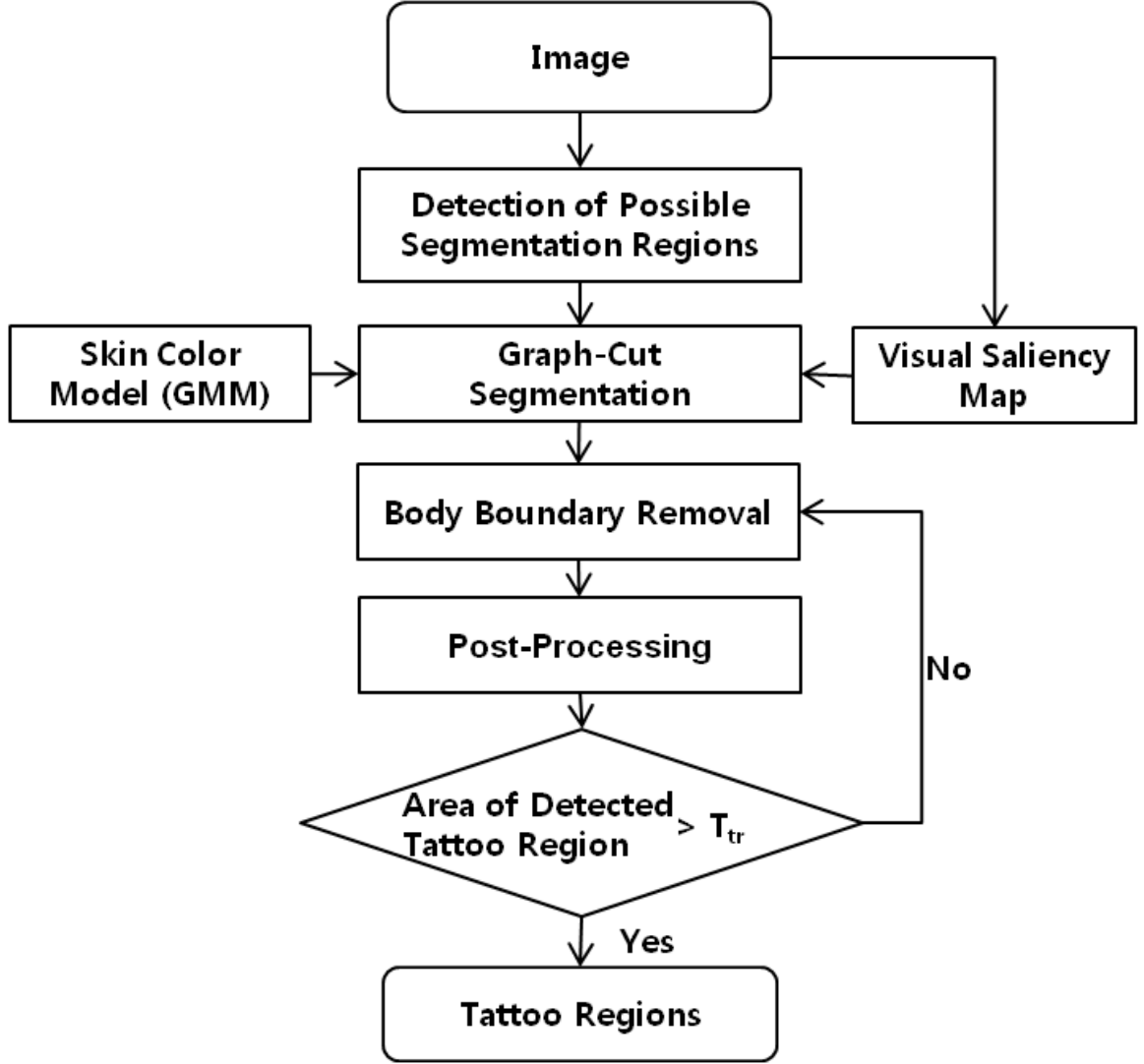


Fig. 2.5.: Our tattoo segmentation system with BBR

where N_m is the number of bin in $h_i(m)$. If (2.11) is satisfied, W_i is determined to have simple texture.

$$\max(h_i(m_{\max}) + h_i(m_{\max} + j)) > t_{st} \quad (2.11)$$

where $\max()$ is a maximum operation, t_{st} is the threshold to detect simple textures and $j = -1, +1$. If we decide W_i has simple textures, the pixel i is removed from the segmented pixels. The removal process is performed for all the segmented pixels.

It is difficult to determine t_{st} because its value that discriminates the boundaries of body and tattoo are different depending on the tattoo image. Large t_{st} does not remove body boundary properly, but small t_{st} can remove the tattoo boundary. Therefore, we iteratively change t_{st} until the area of a detected tattoo region is greater than a pre-defined threshold, T_{tr} . Note that the body boundaries not removed from BBR result in false contours that will be removed in post processing. We can predict if BBR works properly by checking the area of a detected tattoo region after post processing. The tattoo segmentation process with BBR is shown in Figure 2.6.

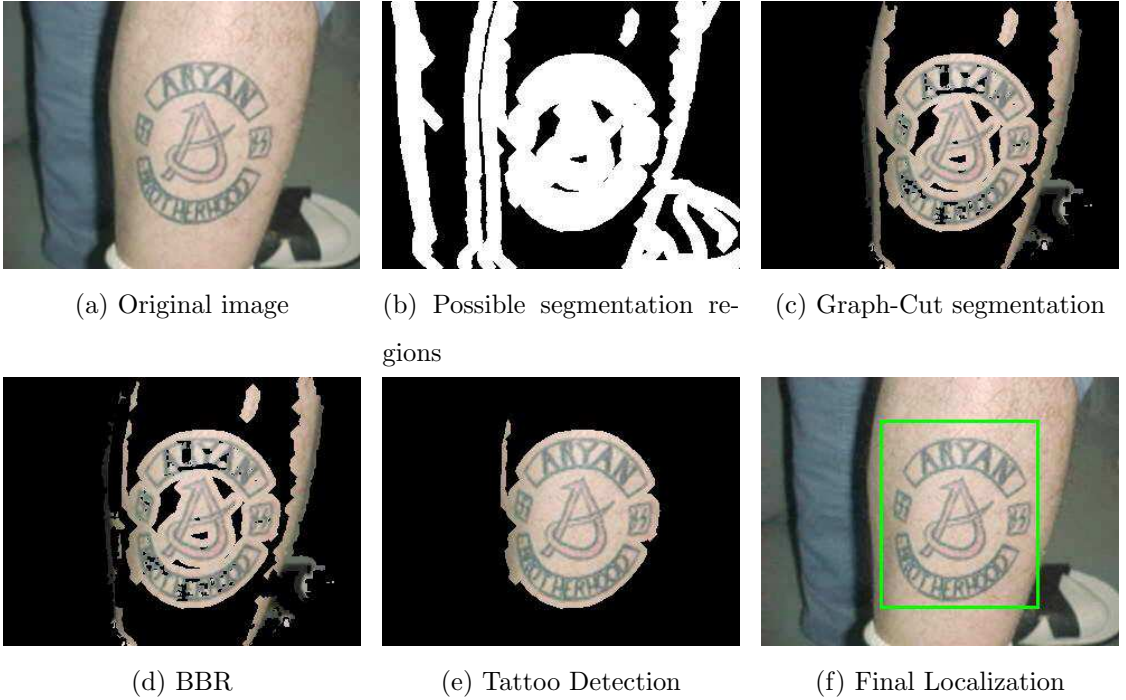


Fig. 2.6.: Graph cut segmentation with BBR

2.4 Experimental Results

2.4.1 Tattoo Segmentation

To evaluate our segmentation method the tattoo images acquired from the Indiana Gang Network (INGangNetwork) are used. Some of these images are shown in Figure 2.8 and Figure 2.9. Since we do not have ground truth for tattoo segmentations we show that our proposed methods can segment tattoo regions well in several images by showing several segmentation results. Also, we compared our methods with [5] and [2]. In our experiments we used the TDSD dataset [87] which includes diverse skin images to train the GMM for our skin color model. The mixture number of the GMM, $M=5$ is used for our experiments. Since the purpose of the tattoo segmentation is to extract tattoo regions from a tattoo image for an image retrieval system, our method does not segment the tattoo inside the closed contour we found after post processing. The tattoo segmentation results of our proposed methods, [5], and [2] are shown in Figure 2.8 and Figure 2.9. As depicted in Figure 2.8 and Figure 2.9, our proposed methods segmented most of tattoo images correctly while [5] made segmentation errors in several tattoo images. Compared to [2] that segments tattoo regions more correctly than [5], our methods (both without BBR and with BBR) still made more correct segmentations. Even though images used in our experiments have diverse skin tones, our methods correctly segmented tattoo regions from the images. When some images have skin-colored background with strong edges, our method can still segment tattoo region correctly. However, some segmentation results showed the errors depicted in Figure 2.7. When tattoo regions are very close to boundaries of the body, our method (without BBR) found the skin pixels connected to both the tattoo regions and the boundaries of the body together, so our post processing could not extract a tattoo region only. Also, if there are hair regions with strong edges inside a body, our method also detected the hair regions as tattoo regions. Our method with BBR solves the problem in the case that tattoo regions are very close to boundaries of the body by removing body boundaries first. However, the errors caused by hair

regions still appear even when we use the method with BBR. This problem will be addressed in future work.



Fig. 2.7.: Incorrect tattoo image segmentation

2.4.2 Tattoo Localization

The goal of this experiment is to find minimum box regions including tattoos in an image instead of segmenting tattoo objects to evaluate our segmentation methods quantitatively. The example of this is depicted in Figure 2.6(f). Two datasets are used for this experiment. Dataset 1 consists of 6308 tattoo images acquired from

the NIST Tattoo Challenge (Tatt-C) dataset [13]. Dataset 2 consists of 1105 tattoo images acquired from the eveiltattoo.com website [61]. To evaluate the performance of our three automatic methods we first manually segment tattoo regions (minimum box regions including tattoos). The manually segmented regions are considered as ground truth. Then, we compute how much the tattoo regions (minimum box regions including tattoos) detected from the proposed automatic methods and the manual segments overlap. We examine the case where the detected tattoo region of the automatic method is larger than that of the ground truth, so we also consider the detected non-tattoo regions for our evaluations. Two evaluation metrics are used.

$$Recall = \frac{n_{OF}}{n_{GDF}} \quad Accuracy = \frac{n_{OF} + n_{OB}}{n_T} \quad (2.12)$$

where n_{OF} is the number of overlapped pixels of the tattoo between automatic detected segments and ground truth, n_{OB} is the number of overlapped pixels of non-tattoo regions, n_{GDF} is the number of pixels in the tattoo in a ground truth image, and n_T is the total number of pixels in a tattoo image. We compute recall and accuracy for each image in a dataset and compute average of them for all the images in a dataset. We compared out three methods: efficient graph-cut without BBR (EGC), efficient graph-cut with BBR (EGCBBR), and center-surround feature based tattoo localization (CSFL) [91,92]. Center-surround feature based tattoo localization (CSFL) was introduced to localize a tattoo region [92]. This method combines a center-surround filter with skin and edge features based on the observation that the skin area surrounding the tattoo is homogeneously smooth and skin-colored.

As shown in Table 2.1, efficient graph-cut with BBR (EGCBBR) is the most accurate among three methods. Then, efficient graph-cut without BBR (EGC) is more accurate than center-surround feature based tattoo localization (CSFL). It is because CSFL did not use visual saliency map and it cannot segment multiple tattoos while both EGCBBR and EGC can do.

Table 2.1.: Recall and Accuracy

	Recall		Accuracy	
	Tatt-C	Evil Tatto	Tatt-C	Evil Tattoo
CSFL	25.18%	42.90%	66.20%	63.26%
EGC	36.16%	63.44%	69.75%	69.30%
EGCBBR	41.25%	66.97%	70.46%	69.91%

2.5 Conclusions and Future Work

In this thesis we described a new tattoo segmentation approach (EGC) by determining skin pixels around a tattoo. Only regions near image edges are considered as possible segmentation regions. In these regions graph-cut segmentation using a skin color model and a visual saliency map was used to find skin pixels. After the graph-cut segmentation we determine which set of skin pixels connected with each other form a closed contour including a tattoo. To remove false contours caused by graph-cut segmentation errors we additionally check the ratio of the area of minimum bounding rectangular box including a closed contour to the area surrounding all pixels connected with the closed contour. The regions surrounded by the final closed contours are considered as tattoo regions. In the experimental results we showed our method achieved more accurate than exiting methods. We also proposed Body Boundary Removal method to be combined with our segmentation method (EGCBBR). By removing body boundary region, we solve the problem of our segmentation method when there are tattoo regions very close to the boundaries of the body. It improved tattoo segmentation accuracies, but it still makes segmentation errors when there are hair regions with strong edges inside a body. Both two methods can segment a tattoo shape boundary as well as localize a tattoo region in an image. For tattoo region localization accuracy, EGCBBR and EGC achieved higher accuracies than CSFL.



Fig. 2.8.: Tattoo image segmentation results

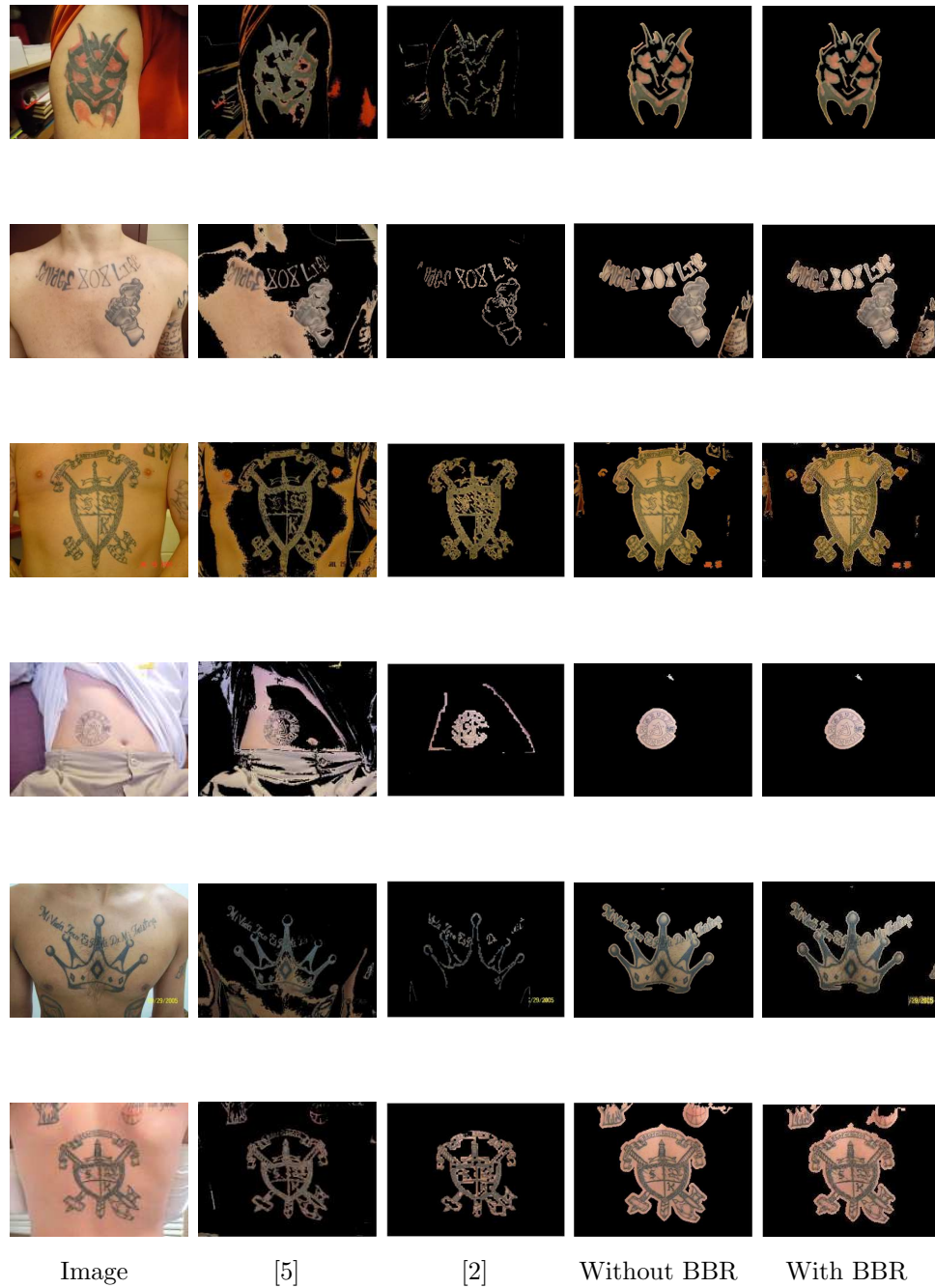
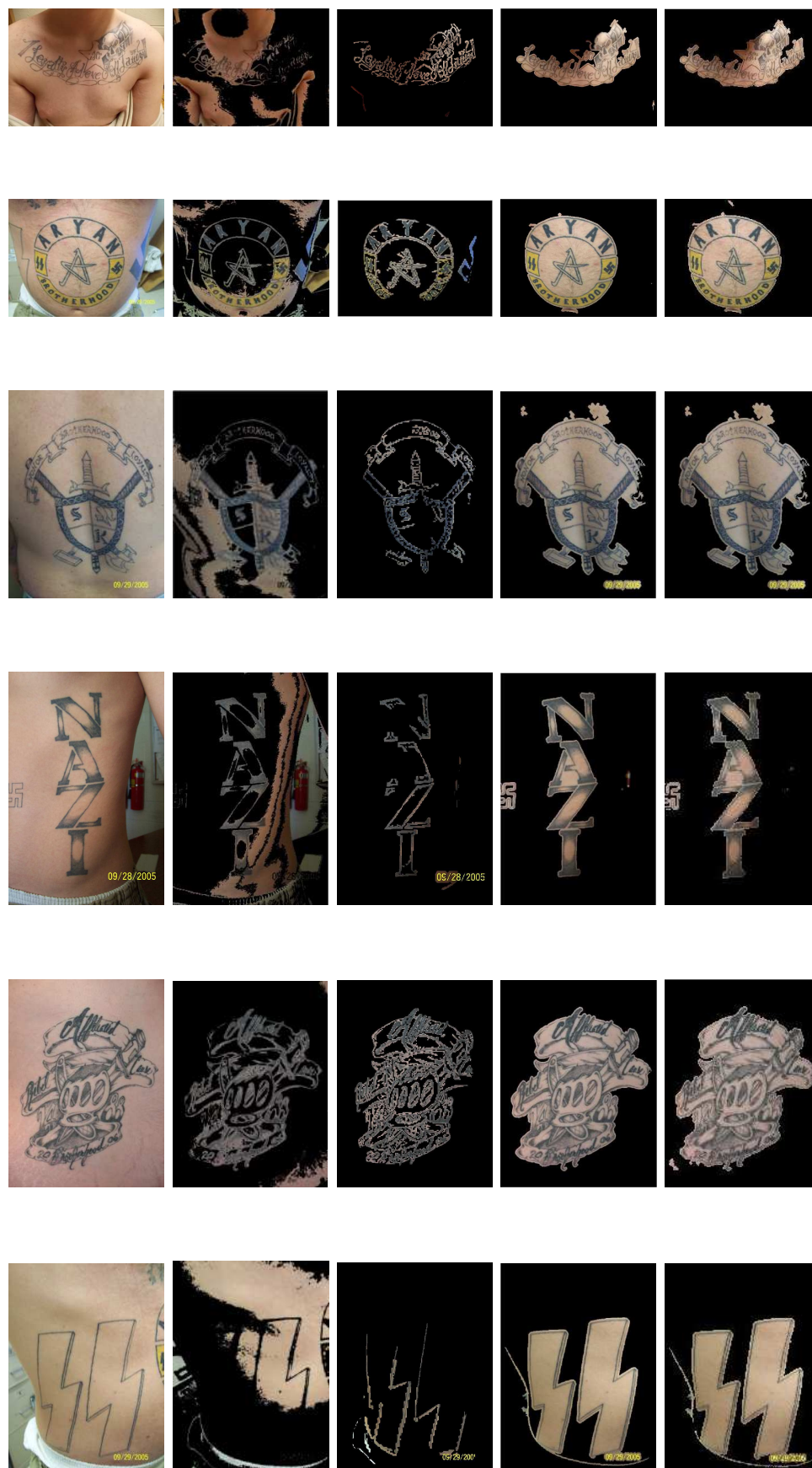


Fig. 2.9.: Tattoo image segmentation results



Image

[5]

[2]

Without BBR

With BBR

Fig. 2.10.: Tattoo image segmentation results

3. TATTOO IMAGE RETRIEVAL BASED ON IMAGE MATCHING

3.1 Review of Existing Methods

There are two important steps in image retrieval based on image matching. One is to match two images via image descriptors robust to several variations such as image deformations, illumination changes, and complex backgrounds. The other one is to define a robust image similarity between two images using the results of the image matching.

There are three steps in image matching. First, image feature points should be extracted first. Interesting points (sparse features) are chosen as image feature points in [93–96] while densely sampled pixels (dense features) are chosen in [62, 97–99]. Depending on the applications, the type of the feature is determined between the sparse feature and the dense feature. The feature descriptors [21, 100–103] for the feature points are then generated using the low level features such as colors, textures, and shapes. The important thing to be considered for constructing these descriptors is the robustness of the descriptors to several image variations such as a pixel illumination change and image deformations. Once the feature descriptors of two images are generated, the feature correspondences between two images are found using an image feature matching.

Once image matching is done, an image similarity between two images is computed using the image matching results, . The image similarity is generally defined based on the distance metric between the image feature descriptors. To compute more robust image similarity, the geometric constraints of the feature points are additionally considered [3, 4, 6, 104, 105]. The image similarity based on the image matching is repeatedly computed on all the pairs of an input image and all the images in a database.

Based on the image similarity scores, the top N matched images are retrieved from a database. The whole process is depicted in Figure 3.1.

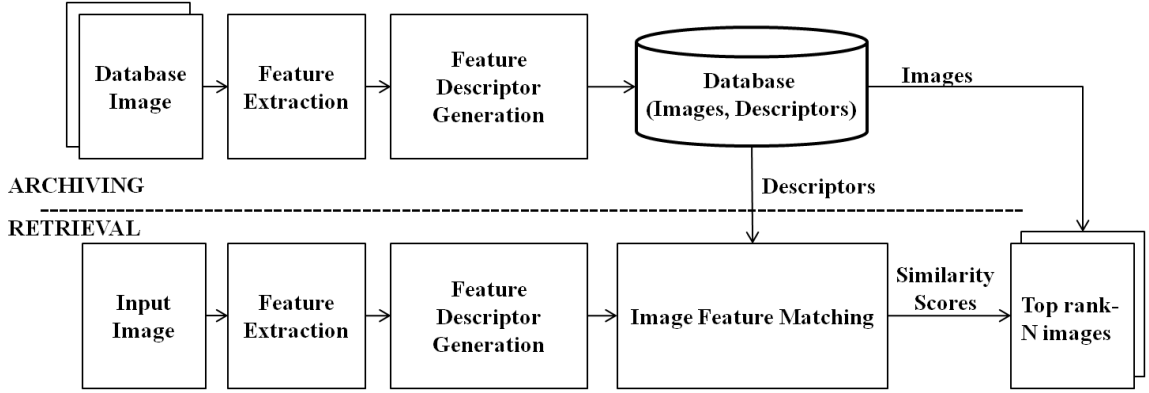


Fig. 3.1.: General image retrieval system based on image matching

Different from the image retrieval based on the image matching using hand crafted features, a deep neural network architecture is also utilized in an image retrieval [34,39,40]. One of the most famous networks for an image retrieval is Siamese network [39]. This network is composed of two independent CNNs. Each CNN accepts an image as an input. Note that all the images should have the same size. The output of each CNN is then compared with each other using a distance metric. By minimizing the difference of the outputs of two CNNs based on the distance metric, the Siamese network is learnt. The whole process is depicted in Figure 3.2.

These image matching and retrieval techniques have also been investigated in the content-based tattoo image retrieval. There exist many methods focused on tattoo image retrieval. In [9] low level features such as color, texture and shape are used for a content-based tattoo image retrieval system. The tattoo regions in an image are first segmented using a morphological operator and color, texture and shape features are extracted from the segmented tattoo regions. A color histogram and a color correlogram [106] are used as color descriptors, and a set of moments invariants is used for a shape descriptor. For a texture descriptor, edge direction coherence vector [107] is used. These descriptors are compared together against the same descriptors of

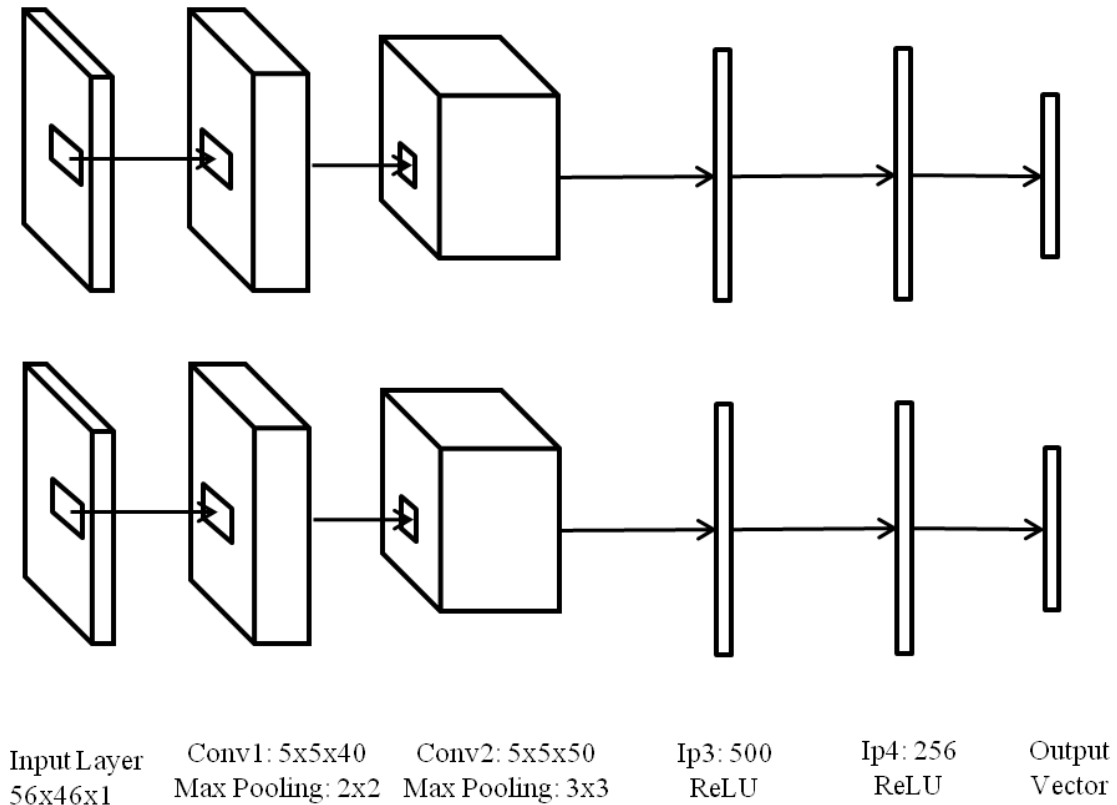


Fig. 3.2.: The Siamese neural network architecture

tattoo images in a database to find the best match. Instead of using a morphological operator, which is not robust to weak edges, active contour based segmentation [18] is used to extract a tattoo from an image in [8]. Using this segmentation approach with global-local (glocal) features the retrieval accuracy was improved. For a global-local feature, a local image edge, the relative positions between the local edge and all the image edges, and color statistics are combined together. In [7] a new rank-based distance metric learning method is described for a tattoo image retrieval system based on low level attributes. With the same feature descriptors in [9], new distance metric between the descriptors of two tattoo images is learnt to improve the retrieval accuracy. In [28] the same authors insists that a tattoo image retrieval system should be based on the concept of “visually similarity,” which can narrow the “semantic gap” [35]. This means there is little connection between pixel statistics and human

interpretation of an image. SIFT (Scale-invariant feature transform) [21] is chosen to find a “visually similar” image, hence improving the image retrieval accuracy with respect to just using low level attributes.

SIFT is also combined with a Bag-Of-Words (BOW) [36] model in [37], [38], and [3]. In [37] the computational complexity of SIFT feature clustering is examined when the features are quantized in the BOW model in large scale image retrieval. Instead of using clustering, a random seed method is proposed to make the quantization process faster and more accurate. In [38] the quantization error caused by feature clustering in the BOW model is further studied using multiple BOW models. The weighted averages of multiple clusters is used to avoid this quantization errors. The Ranking SVM [108] is adopted to learn the weights. The quantization error problem in the BOW model is also addressed in [3]. Hamming embedding and geometry consistency [109] are additionally used to solve the quantization error problem. Using hamming embedding, an image feature vector is converted to a binary image vector. The histogram of the orientation differences of the matched features in two images is used to check geometry consistency. Instead of SIFT, SURF(Speeded Up Robust Features) [100] is used for tattoo image matching in [110]. They compared SURF and MU-SURF (Modified Upright SURF) [111] on tattoo image datasets with several conditions (rotation transformation, RGB noise insertion, cropped images), and they showed that MU-SURF was more accurate than SURF. The main difference between MU-SURF and SURF is that MU-SURF uses the larger size of descriptor window and the subregions for computing Haar wavelet responses. In [112] Exemplar Codes [113] are used with linear SVM classifiers [114] for tattoo image classification. Each exemplar classifier is trained for each class of an object first. The EVT (extreme value theory) normalization [115] for one exemplar becomes one feature vector for the final SVM classifier. They show that their method can classify a tattoo object faster than the method of [113] with similar accuracy.

In [4,6,29–32] various matching based tattoo image retrieval systems are described. Matching a tattoo sketch used in the identification of a suspect with a real tattoo

image is discussed in [29, 31]. In the both methods, edge features are mainly chosen to match the two different types of images. In [29] SIFT descriptors are extracted from both a tattoo sketch and an edge image, and Sparse Representation-based Classification (SRC) [58, 116] is effectively utilized to match these descriptors. Instead of SIFT, Shape Context (SC) based shape descriptor [117] is used with edges in [31]. The feature correspondences via shape matching based on SC are additionally refined using the point alignment technique called Coherent Point Drift (CPD) [118].

Geometric constraints of SIFT are used to improve image matching accuracy in [6]. The SIFT feature matching is done first to find the all the feature correspondence between an input image and all the images in a database. Based on an observation that the feature point in an input image matched to many other feature points in all the database images is likely to cause false matching, the feature correspondences obtained by the SIFT feature matching are refined. The number of the refined feature correspondences is used as an image similarity between two images. In [4] a robust similarity metric is described for SIFT based image matching. The robust similarity metric based on relationship between the matched SIFT feature points of tattoo images improves image retrieval accuracy. To generate the robust similarity there are two main observations. First one is the same observation about false matching in [6]. The other one is that the multiple feature points in an input image matched to the one feature point in the database image are likely to cause false matching as well. By assigning weak weights to the features that can cause false matching, the robust image similarity metric is defined. The additional use of metadata from tattoo images (i.e., text identifying the tattoo) is also used to improve accuracy and reduce computational complexity. In [32] the advanced feature extraction method is addressed. Using higher order scale space, the image matching more robust to scale deformation is achieved. In [30] they show that the image registration between two tattoo images can improve the image recognition accuracy. To find feature correspondences SIFT descriptors are used with the RANSAC [119] first. Once the feature correspondences are found, images are registered. The registration evaluation process is then used to check if

there are registration errors. If the registration errors are detected, a registration error correction is additionally used and the final image matching is executed.

There have been lots of deep neural network learning methods to recognize a tattoo object [15, 17, 33, 34]. In [33], CNN (convolutional neural networks) [26] is customized to determine if an image includes a tattoo. This network consists of five convolutional layers and three fully connected layers. Since this network needs to classify a tattoo image or a non-tattoo image, the last fully connected layer has only two neurons: one for tattoo and the other for non-tattoo. The experiment shows that the CNN based tattoo classification outperforms all the methods reported in NIST tattoo challenge [11]. The correlation neural network is used to classify the tattoo object after the tattoo localization in [15]. This network consists of four layers: the first layer is an input layer, the second layer is a fully connected layer, the third layer is a fully connected layer but it has less number of neurons than the second layer, and the last layer is an output layer whose a neuron represents a class of similar tattoos. In [34], they use Siamese network [39] for matching two tattoo images and convolutional neural network (CNN) for classifying a tattoo image. This Siamese network consists of two CNNs. Each of an input image and a reference image is presented into each CNN. The output of each CNN is then compared with each other using the triplet loss function [41]. By minimizing the loss function, the Siamese network is learnt. The experiment shows that the Siamese network based image matching method outperforms all the methods reported in [11] for the tattoo similarity dataset [13]. The faster R-CNN [24, 27] is used to localize tattoo regions and classify the tattoos at the same time in [17]. The region proposal [42–45] that can include tattoo objects is extracted first. The extracted region proposal is presented into CNN as an input image. The object class vector is generated as the output of the R-CNN, and the class with the maximum element of the vector is chosen as the class of the object in the region proposal.

3.2 Tattoo Image Retrieval System Based On Multiple Histograms Based Local Context (MHLC) Descriptor

In this section, our tattoo image retrieval system is described. Our contribution is the creation of new local and global shape descriptor robust to scale, translation, rotation, and shape distortions. By using the scale invariant feature transform (SIFT) with local shape context based on multiple different sized-bin polar histograms (MH), more accurate image matching can be obtained. A global shape descriptor based on MH and a 2D Fourier Transform is also used for robustness of translation, scale, rotation and shape distortions. We also describe robust similarity for local descriptors and a weighted matching method based on local and global descriptors.

3.2.1 System Overview

Figure 3.3 shows the block diagram of our proposed tattoo image retrieval system. The system is divided into two parts. In the archiving process we first extract SIFT features from each image in our database. From these features we generate a set of multiple different sized-bin histograms based local context (MHLC) descriptors and a global shape descriptor. For MHLC descriptor local shape contexts based on multiple different sized-bin polar histograms are combined with SIFT descriptors. The global shape descriptor is based on the locations of all the SIFT features in the image. The two types of descriptors are stored in the database. Therefore our database consists of images and descriptors. Given an input image that we would like to compare to our database for similar images we first extract SIFT features from the image and then we generate a set of MHLC descriptors and a global shape descriptor. We then compare these features from the image against the images in our database using two different matching methods for the two types of descriptors. Each of the two matching methods returns one score which we combine into one weighted score to retrieve the top N matched images.

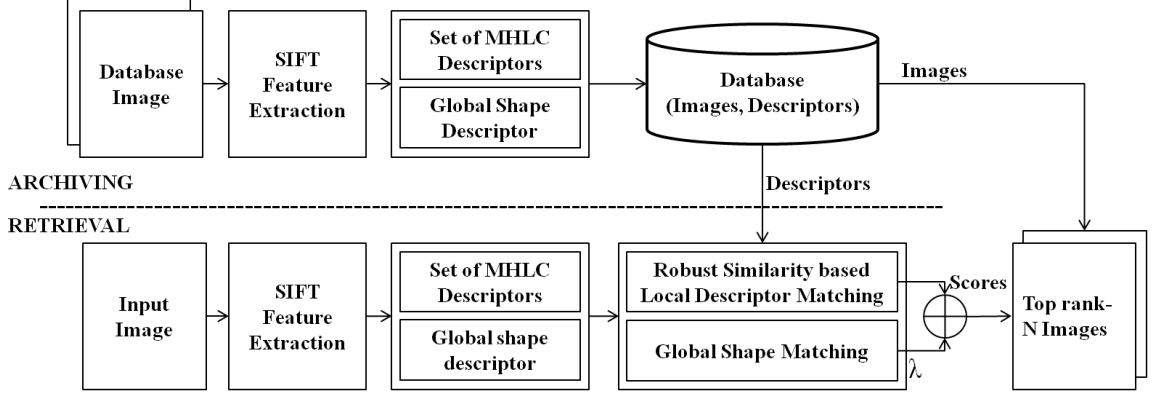


Fig. 3.3.: Tattoo image retrieval system based on MHLC descriptor

3.2.2 Multiple Different Sized-Bin Histograms based Local Context (MHLC) Descriptor

SIFT has been widely used in object recognition because SIFT features are invariant to scale and rotation and are robust to affine deformation, illumination change, and change in 3D viewpoint [21,63,97]. Since SIFT has proved to be effective in other tattoo image retrieval applications [3,4,29], we also used SIFT in our system. Image matching using robust similarity measure for SIFT descriptors in [4] achieved high matching accuracy between near duplicate tattoo images taken from the same subject in different environments. However, it becomes problematic in image matching between images taken from different subjects who have similar tattoo object shapes because a SIFT descriptor itself lacks of spatial information. In [120] an image matching method combining SIFT descriptors and global contexts was presented. They described a global shape context combined with a SIFT descriptor for a feature point based on spatial locations of all SIFT feature points relative to the feature point. Since, however, most tattoo images have partial shape distortion, the global shape context based on the spatial locations of all feature points relative to a feature point cannot describe the tattoo object shapes properly. Also, they used curvature values on the SIFT feature point locations to generate the global shape context, which are not consistent on the tattoo images that have similar tattoo object shapes but come

from different subjects. To address this problem, we use a local shape context (LC) that considers the spatial locations of neighbor features only. Additionally, we propose a local shape context based on multiple different sized-bin histograms (MHLC) for a SIFT feature. If the size of bin is very small, the shape context can represent a shape of an object in detail, but it cannot represent distorted shapes properly. If the size of bin is very large, the shape context can represent the distorted shape properly, but it cannot represent the detailed shapes of objects well. To solve this problem we compute multiple polar histograms whose bins' sizes are changed from small to large and combine them together to generate a local shape context. After generating the local shape context, we generate our proposed MHLC descriptor by combining a SIFT descriptor and MHLC. To describe the proposed local shape context, we need to define neighbors of a feature point first. To define the neighbors invariant to image scale we first normalize all SIFT feature point locations relative to the centroid f_c of all the feature points. The f_c is computed as:

$$f_c = \frac{1}{N_f} \sum_{i=1}^{N_f} f_i \quad \text{for } i=1,2,..N_f \quad (3.1)$$

where f_i is the spatial location of the i^{th} SIFT feature point and N_f is the total number of SIFT feature points in the image. Then we compute the distances and the angles between all f_i and f_c :

$$r_{ic} = ||f_i - f_c||_2, \quad \theta_{ic} = \arctan \frac{f_{iy} - f_{cy}}{f_{ix} - f_{cx}} \quad (3.2)$$

The normalized i^{th} feature point relative to the centroid, f_{norm_ic} is computed as:

$$f_{norm_ic} = \frac{1}{r_{mn_c}}(f_{icx}, f_{icy}) = \frac{1}{r_{mn_c}}(r_{ic}\cos\theta_{ic}, r_{ic}\sin\theta_{ic}) \quad (3.3)$$

where r_{mn_c} is the average of r_{ic} over all i . All the j^{th} feature points to satisfying (3.4) are then considered to be neighbors of i^{th} feature point.

$$||f_{norm_jc} - f_{norm_ic}||_2 \leq r_{th} \quad \text{for } j = 1, 2, ..N_f, j \neq i \quad (3.4)$$

where $r_{th}=2.0$ is a distance threshold to define neighbors of a feature point. We call B_i the neighbors of the i^{th} feature point and N_{B_i} the number of the neighbors. Once the neighbors of each feature point are determined, the distances and the angles between all f_j and f_i for $j \in B_i$ are computed as:

$$r_{ji} = \|f_j - f_i\|_2, \quad \theta_{ji} = \arctan \frac{f_{jy} - f_{iy}}{f_{jx} - f_{ix}} \text{ for } j \in B_i \quad (3.5)$$

Then, we compute a 2D histogram, h_{m_r, n_θ}^i that represents the 2D spatial distribution of neighbors of the i^{th} feature point:

$$\begin{aligned} h_{m_r, n_\theta}^i(m, n) &= \sum_{j \in B_i} \#(r_l \leq \frac{r_{ji}}{r_{mn_i}} < r_u, \theta_l \leq \theta_{ji} - \theta'_i < \theta_u) \\ r_l &= \frac{r_{max}}{m_r}(m-1), r_u = \frac{r_{max}}{m_r}m \text{ for } m = 1, \dots, m_r \\ \theta_l &= \frac{2\pi}{n_\theta}(n-1), \theta_u = \frac{2\pi}{n_\theta}n \text{ for } n = 1, \dots, n_\theta \end{aligned} \quad (3.6)$$

where $h_{m_r, n_\theta}^i(m, n)$ is the 2D polar histogram centered on f_i , m_r is a parameter of the histogram for the number of bins for the radius, n_θ is a parameter of the histogram for the number of bins for the angles, $r_{max}=2$ is a constant for maximum limit of a distance, θ'_i is the dominant local orientation obtained from the i^{th} SIFT feature, r_{mn_i} is the average of r_{ji} over all $j \in B_i$, and $\#()$ is a counting operator. Since we normalize r_{ji} by r_{mn_i} and align all θ_{ji} by θ'_i , the local shape context based on h_{m_r, n_θ}^i is invariant to translation, scale and rotation. We combine then multiple 2D polar histograms whose bins' sizes are changed from small to large for our local shape context to be robust to shape distortions as well as to represent detailed shape information.

$$lsc^i = \frac{1}{N_{B_i}} \begin{bmatrix} lsc_1^i \\ lsc_2^i \\ lsc_3^i \end{bmatrix} = \frac{1}{N_{B_i}} \begin{bmatrix} vec(h_{m_r, 0.5n_\theta}^i) \\ vec(h_{m_r, n_\theta}^i) \\ vec(h_{m_r, 2n_\theta}^i) \end{bmatrix} \quad (3.7)$$

where $\text{vec}()$ is the vectorization operator, lsc_1^i , lsc_2^i , and lsc_3^i are vectorized histogram whose angle parameters are $0.5n_\theta$, n_θ , and $2n_\theta$ respectively. Next, the i^{th} local shape context, lsc^i is normalized to be combined with a SIFT descriptor.

$$nlsc^i = \frac{lsc^i}{\|lsc^i\|_2} \quad (3.8)$$

where $nlsc^i$ is the normalized local shape context. Finally, our MHLC descriptor for i^{th} feature point, $MHLCD^i$ is created by :

$$MHLCD^i = \begin{bmatrix} w_1 Sd^i \\ w_2 nlsc^i \end{bmatrix} \quad (3.9)$$

where Sd^i is the normalized i^{th} SIFT descriptor and $w_1=0.5$ and $w_2=0.5$ are the weights corresponding to Sd^i and $nlsc^i$. Since a SIFT descriptor and h_{m_r, n_θ}^i are invariant to translation, scale, and rotation, our $MHLCD^i$ is also invariant to them. For the MHLC descriptor, $m_r=3$ and $n_\theta=12$ are used in this thesis.

3.2.3 Global Shape Descriptor

Even though the local spatial distribution of feature points is more robust to represent partially distorted shapes, a global spatial distribution is also a good resource to represent a tattoo object shape. For this reason, a global shape descriptor using the multiple 2D histograms based shape context similar to MHLC, is proposed. Different from MHLC descriptors, one global shape descriptor is generated for one image. Let h_{m_r, n_θ}^c be a histogram which represents the spatial distribution of all feature points relative to their global centroid. Then, the neighbors of the centroid, B_c are defined as all SIFT feature points. However, the dominant local orientation (θ'_c) for rotation invariance, is not defined for the centroid. Therefore, we propose a method to make our global shape descriptor robust to rotation. First, θ'_c is estimated based on the counts of SIFT feature locations around a set of fan-shaped windows. Figure 3.4(a) and 3.4(d) illustrate these windows in blue overlaid on all feature points. The windows are separated by $\theta_{bin} = 2\pi/n_\theta$, which is the size of each bin in angular direction.

While rotating the set of windows from 0 to θ_{bin} together we count the number of feature points inside the windows. The orientation at which the count is maximum is considered as the reference-axis, θ'_c . That is,

$$\theta'_c = \underset{j}{\operatorname{argmax}} W(\theta_{jc}) \quad \text{for } \theta_{jc} \in [0, \theta_{bin}) \quad (3.10)$$

where $W(\theta_{jc})$ is number of feature points inside a set of windows rotated by θ_{jc} . Note that because of the angular bins being θ_{bin} from each other any rotation by $\theta_{bin}n + (\pi/180, 2\pi/180, \dots)$, $n = 0, 1, \dots$ will produce the same bin count distribution, but with a circular displacement. Figure 3.4 illustrates this property. Therefore, we can use the magnitude of the 2D Fourier Transform of the histogram to make the descriptor rotation invariant. Our global shape descriptor $GSD(u, v)$ is then defined as

$$GSD_{m_r, n_\theta}(u, v) = |DFT\{\frac{h_{m_r, n_\theta}^c(m, n)}{N_f}\}| \quad (3.11)$$

$$m, u = 1, 2, \dots, m_r \quad n, v = 1, 2, \dots, n_\theta,$$

where $DFT()$ is the 2D DFT operator. Similar to MHLC, our global shape descriptor is generated by combining multiple different sized-bin histograms together to be robust to partial shape distortions.

$$VGSD_{final} = \begin{bmatrix} VGSD_1 \\ VGSD_2 \\ VGSD_3 \end{bmatrix} = \begin{bmatrix} \text{vec}(GSD_{m_r, 0.5n_\theta}) \\ \text{vec}(GSD_{m_r, n_\theta}) \\ \text{vec}(GSD_{m_r, 2n_\theta}) \end{bmatrix} \quad (3.12)$$

where GSD_1 , GSD_2 , and GSD_3 are shape descriptors whose angle parameters are $0.5n_\theta$, n_θ , and $2n_\theta$ respectively. For our global shape descriptor, $m_r=7$ and $n_\theta=12$ are used in this thesis. Then, $VGSD_{final}$ is a 147-dimensional vector.

3.2.4 Image Matching

In our tattoo image retrieval system we use two different image matching methods: one for the MHLC descriptors and one for the global shape descriptor. For MHLC

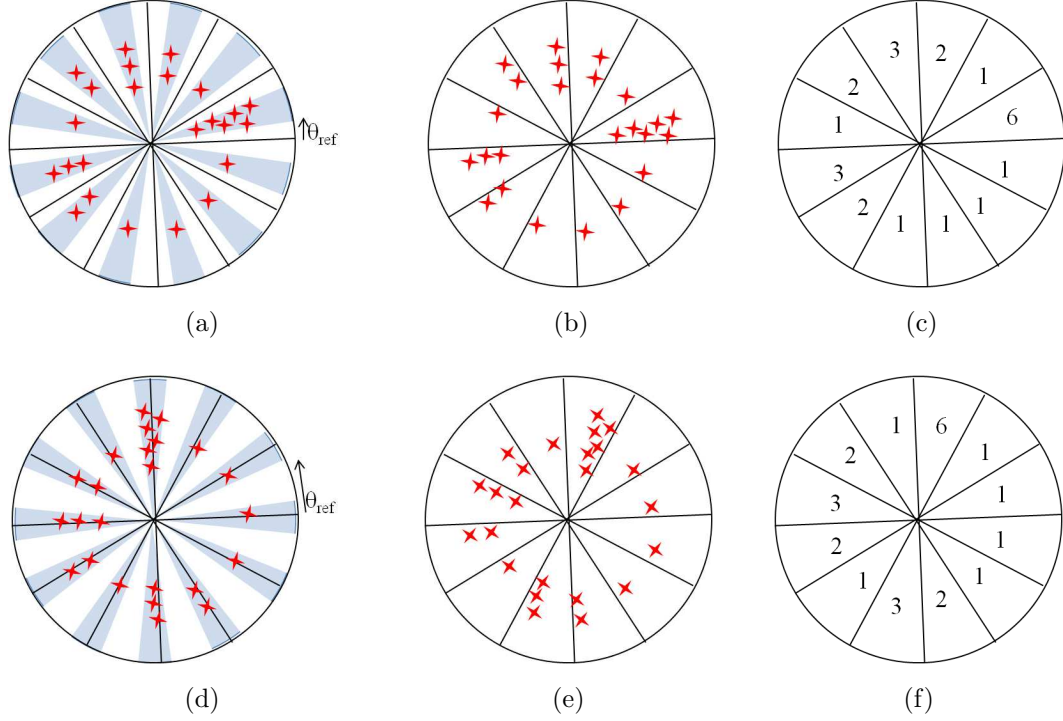


Fig. 3.4.: Example of computing θ'_c and the polar histogram: In (a) the orientation with maximum feature location count is $\theta'_c = \pi/18$. In (d) features are rotated $8\pi/18$ with respect to (a), yielding $\theta'_c = 2\pi/18$. (b) and (e) show the aligned feature locations. The histograms computed in (c) and (f) have the same count distribution, but with a circular displacement

descriptor matching we use a robust similarity metric, S_W , which assigns low weight to indistinct features.

$$S_W(I_q, I_k) = \sum_{i=1}^{N_f} x_i \left(\frac{1}{m^i(I_k)} \log \frac{N_G}{n^i} \right) \quad (3.13)$$

where I_q is an input image, I_k is the k^{th} database image, $m^i(I_k)$ is the number of feature points including the i^{th} feature point in I_q that are matched to the same feature point in I_k , n^i is the number of database images that have feature points matched to i^{th} feature point in I_q , and N_G is the total number of database images. x_i is 1 if the ratio of the closest match to the second closest match between $MHLC D^i$ in I_q and all $MHLC D^j$ in I_k is less than pre-defined threshold, $t_p = 0.69$. Otherwise,

x_i is 0. The robust similarity is also used with SIFT descriptors in [4], but we use it for our MHLC descriptors here. For global shape descriptor matching we define a similarity metric S_G as

$$S_G(I_q, I_k) = -\frac{1}{2} \sum_{i=1}^{147} \frac{(VGSD_{final}^q(i) - VGSD_{final}^k(i))^2}{VGSD_{final}^q(i) + VGSD_{final}^k(i)} \quad (3.14)$$

$$k = 1, 2, \dots, N_G,$$

where VSD_{final}^q is the final global shape descriptor for I_q and VSD_{final}^k is the final global shape descriptor for I_k . For retrieval, we choose the most similar tattoo images in the database to maximize total similarity, S_T , defined as:

$$S_T(I_q, I_k) = S_W(I_q, I_k) + \lambda S_G(I_q, I_k) \quad (3.15)$$

where λ is a weight to compensate the fact that $S_W(I_q, I_k)$ is based on feature count and $S_G(I_q, I_k)$ is based on a distance metric. In our experiments $\lambda=36$ produced the best results.

3.2.5 Experimental Results

In this experiment three different datasets are used to evaluate our tattoo image retrieval system. Dataset 1 consisted of 392 tattoo images acquired from the Indiana State Police: 101 input images, 123 database images, and 168 background images. Note that background image is an image that is added to the database images as “distractor” to confuse the retrieve process Dataset 2 consisted of 1493 tattoo images acquired from eviltattoo.com [61]: 481 input images, 622 database images, and 390 background images. Dataset 3 consisted of 2212 tattoo images acquired from eviltattoo.com [61]: 851 input images and 1361 database images. Note that all the images in dataset 1, 2, and 3 are manually cropped to contain tattoo areas only. The description of our datasets is summarized in Table 3.1. To evaluate the performance of our proposed method we used the Cumulative Match Characteristic (CMC) [121] to obtain the top-N (N=20) rank retrievals from our database.

Table 3.1.: The description for our datasets

	Number of Input Image	Number of Database Image	Number of Background Image	Number of Different Tattoo Object	Cropped
Dataset 1	101	123	168	51	Yes
Dataset 2	481	622	390	21	Yes
Dataset 3	851	1361	0	851	Yes

Indiana State Police Dataset (Dataset 1)

All the tattoo images in this dataset are manually cropped to contain tattoo areas only. Most images in this dataset are images which have the same tattoo objects, but come from different subjects. Therefore, there are lots of shape distortions in the same tattoo object images. For this dataset, the image retrieval system illustrated in Section 3.2 is used because a global shape descriptor can be used for cropped images. Figure 3.5 shows some sample images in this dataset.



Fig. 3.5.: Sample tattoo images in this dataset

Table 3.2.: CMC in dataset 1 (*unit : %*)

Method	rank-1	rank-10	rank-20
Proposed(GRMHLC)	70.3	79.21	84.16
Proposed(RMHLC)	67.33	77.23	84.16
Proposed(MHLC)	58.42	74.26	84.11
Proposed(HLC)	56.44	73.27	81.19
RIS [4]	55.45	77.23	83.17
SIFT+SC [117]	59.41	77.23	83.17
SIFT+GC [120]	39.6	65.35	75.25
SIFT [21]	46.53	53.47	66.34

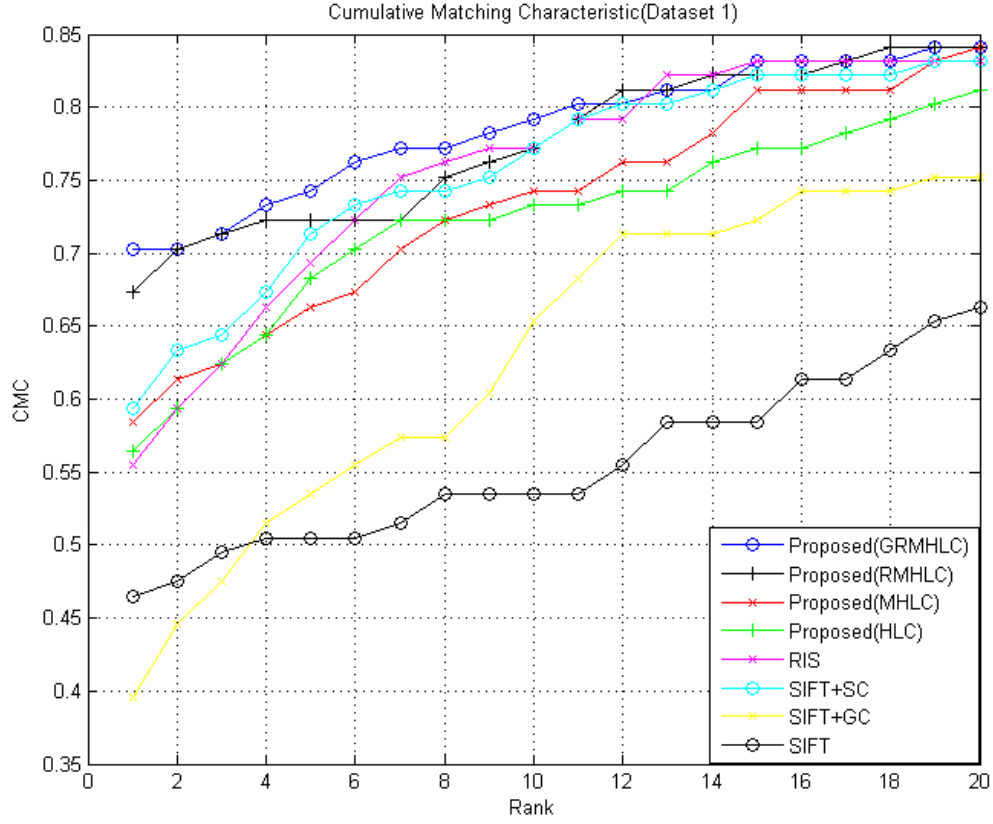


Fig. 3.6.: Retrieval performance comparison of our proposed methods, [4], SIFT+ [117], [120], and [21] using CMC in Dataset 1

We compared our method against RIS [4], SIFT+SC [117], SIFT+GC [120], and SIFT [21] in terms of the CMC by implementing their system and testing them on dataset 1. In RIS robust image similarity based on SIFT descriptor is used. SIFT+SC is the combination of SIFT and the shape context proposed in [117]. SIFT+GC is the combination of SIFT and the global context proposed in [120]. Proposed(HLC) is the combination of SIFT and a local shape context based on one fixed bin-sized histogram, Proposed(MHLC) is our method using MHLC descriptor without robust similarity and global shape descriptor, Proposed(RMHLC) is the Proposed(MHLC)

combined with robust similarity, and Proposed(GRMHLC) is the Proposed(RMHLC) combined with our global shape descriptor.

Figure 3.6 and Table 3.2 show the experimental results. The experimental results demonstrate that the Proposed(MHLC) is more accurate than the Proposed(HLC). They also shows that our MHLC with robust similarity can improve tattoo image retrieval accuracy a lot. Even though our global shape descriptor did not improve the accuracy a lot compared to the improvement of RMHLC, it also shows that a global spatial distribution of features is a useful resource, especially to improve top rank-1 accuracy. As shown in Table 3.2 our Proposed(GRMHLC) achieves 70.3% top rank-1 accuracy and 79.21% top rank-10 accuracy in dataset 1. It outperforms RIS [4] by 13.3% in top rank-1 in dataset 1.

Eviltattoo Dataset (Dataset 2)

All the tattoo images in the dataset 2 are also manually cropped to contain tattoo areas only. Compared to dataset 1, there are more tattoo images in the dataset 2. Also, it has more tattoo images that have the same tattoo objects but come from different subjects than dataset 1. Therefore, there are more shape variations in the same tattoo object images in the dataset 2. Since all the tattoo images in dataset 2 are also manually cropped, the system illustrated in Section 3.2 is also used here.

For the dataset 2, we also compared our method against RIS [4], SIFT+SC [117], SIFT+GC [120], and SIFT [21] in terms of the CMC. Figure 3.7 and Table 3.3 show the experimental results for the dataset 2. The improvement of the retrieval accuracy using MHLC descriptor is more obvious in the dataset 2 than in the dataset 1. (4.15% more than HLC for rank 1 in dataset 2 and 1.98% more than HLC for rank 1 in dataset 1). That demonstrates that our MHLC descriptors are more powerful on the images that have shape distortions or variations of the same tattoo objects.

This experimental results also show that the retrieval accuracy is improved a lot when our MHLC descriptor is used with robust similarity metric. With additional

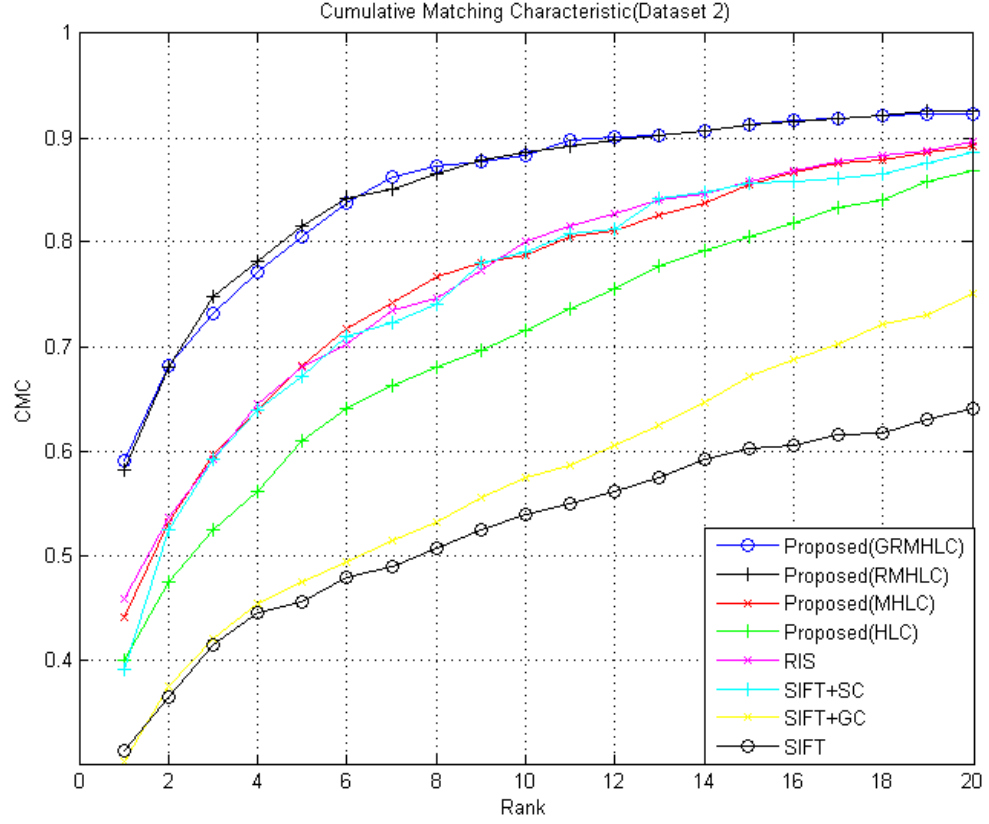


Fig. 3.7.: Retrieval performance comparison of our proposed methods, [4], SIFT+ [117], [120], and [21] using CMC in Dataset 2

use of our global shape descriptor, our method achieves 59.84% top rank-1 accuracy and 88.36% top rank-10 accuracy in dataset 2. It outperforms RIS [4] 14.85% in top rank-1 in dataset 2.

NIST Tattoo Similarity Dataset (Dataset 3)

In this experiment, tattoo similarity dataset (dataset 3) is also used to evaluate our tattoo image retrieval system. This dataset is obtained from NIST tattoo challenge dataset [13]. The NIST tattoo similarity dataset consists of 2212 tattoo images: 851 input images, 1361 database images. The description of our datasets is summarized in

Table 3.3.: CMC in dataset 2 (*unit : %*)

Method	rank-1	rank-10	rank-20
Proposed(GRMHLC)	59.04	88.36	92.31
Proposed(RMHLC)	58.21	88.57	92.52
Proposed(MHLC)	44.07	78.79	89.19
Proposed(HLC)	39.92	71.52	86.9
RIS [4]	45.74	80.04	89.6
SIFT+SC [117]	39.09	79.00	88.57
SIFT+GC [120]	30.15	57.38	75.05
SIFT [21]	31.19	53.85	64.03

Table 3.1. To evaluate the performance of our proposed method the same evaluation protocol provided in [11, 34] is used. The tattoo similarity dataset is split into 5 sub-datasets for 5-fold cross validation. The CMC score is then computed for each sub-dataset and the average CMC score is obtained. We compared our method against the methods described in [11, 34] in terms of the average CMC. Different from our previous experiments, we compared our method with deep learning based image retrieval [34] as well.

As shown in Table 3.4, our experimental results show that our RMHLC is more accurate than the deep learning based image retrieval method in [34] as well as all the methods reported in [11]. Even though RMHLC is slightly less accurate than Deep Tattoo 2 in top rank-1 by 0.4%, it is much more accurate than Deep Tattoo 2 in top rank-10 and top rank-20 by 5.7% and 4.9% respectively.

Table 3.4.: CMC in dataset 3 (*unit : %*)

	CMC		
Method	rank-1	rank-10	rank-20
Compass [11]	0.5	7.4	14.7
MITRE [11]	3.5	14.9	23.9
Deep Tattoo 1 [34]	1.7	11.1	15.5
Deep Tattoo 2 [34]	5.5	16.4	24.9
RMHLC	5.1	22.1	29.8

3.3 Tattoo Image Retrieval System Based On Dense Multiple Histograms Based Local Context (DMHLC) Descriptor

In this section, we introduce our modified tattoo image retrieval system. This system is almost the same as the system described in Section 3.2. Different from our previous system that uses the MHLC image descriptor, dense multiple histograms based local context (DMHLC) descriptor is used in this system. The MHLC descriptor represents the shape of a tattoo object using the spatial distribution of the SIFT features. However, it sometimes fails to represent the shape of a tattoo object correctly when the SIFT feature points are not densely distributed on the tattoo object. To solve this problem, a DMHLC descriptor uses the spatial distribution of the densely sampled features on the tattoo object instead. Except using the multiple histograms based local context on densely sampled features, the descriptor generation is the same as the the generation of the MHLC descriptor.

3.3.1 System Overview

Figure 3.8 shows the block diagram of tattoo image retrieval system based on the DMHLC descriptor. The system is divided into two parts. In the archiving process we first extract SIFT features from each image in our database. Also densely sampled features on a tattoo object are extracted. From these features we generate a set of dense multiple different sized-bin histograms based local context (DMHLC) descriptors. For the DMHLC descriptor a local shape context based on multiple polar histograms on the dense sampled features is generated and it is combined with a SIFT descriptor. Note that one DMHLC is generated for one SIFT feature point, so the number of the DMHLC descriptors is the same as the number of the SIFT feature points. Our database consists of images and descriptors. Given an input image that we would like to compare to our database for similar images we extract SIFT features and densely sampled features from the image. We generate a set of DMHLC descriptors. We then compare the DMHLC descriptors from the image against the

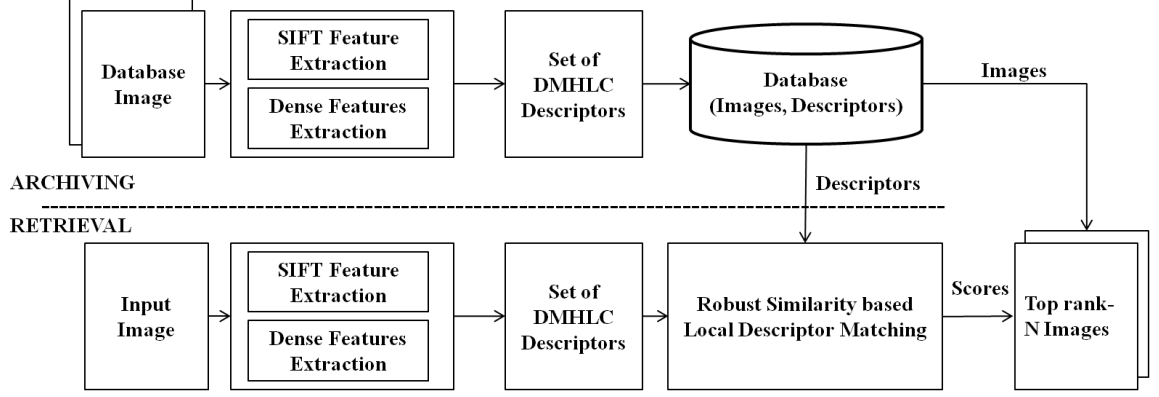


Fig. 3.8.: Tattoo image retrieval system based on DMHLC descriptors

DMHLC descriptors in our database using image matching method. It returns image similarity scores between an input image and a database image to retrieve the top N matched images.

3.3.2 Dense Multiple Different Sized-Bin Histograms based Local Context (DMHLC) Descriptor

The MHLC descriptor consists of two different image descriptors: the SIFT descriptor and the multiple polar histograms based image descriptor. The multiple polar histograms represent the tattoo shape while SIFT descriptor describes the tattoo appearance. These polar histograms are computed on the spatial locations of the SIFT feature points. In the polar histogram based shape descriptor, the uniformly and densely distributed the features on the tattoo are very important. However, the SIFT feature points are not uniformly distributed on the tattoo. Especially, when an image has a low resolution, the image has very small number of the SIFT feature points. In this case the tattoo shape cannot be represent accurately using the MHLC descriptor. To solve this problem, we extract the uniformly sampled features on the tattoo as well as the SIFT feature points. Since the uniformly sampled features should be on the tattoo region only, we filter out some of them that have small gradient mag-

nitudes. The spatial distribution of the filtered out features is computed using the 2D polar local histogram centered on each SIFT feature point. Like MHLC, the multiple histograms that have the different sized bins are generated, and it is combined with the SIFT descriptor to generate final our DMHLC. This process is well depicted in Figure 3.9.

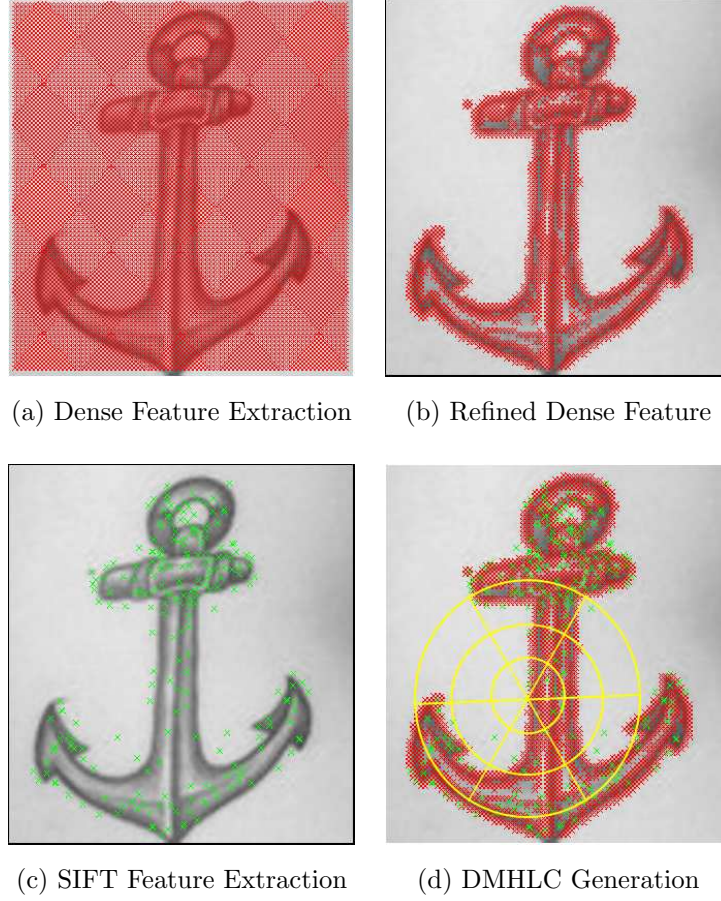


Fig. 3.9.: Example of the DMHLC descriptor generation

To extract the dense features, pixels are subsampled from an image at every 4 pixel along horizontal and vertical axis. To keep the dense features on the tattoo region only, we filter out the dense features that have smaller gradient magnitudes than pre-defined threshold, th_{df} . Let the j^{th} refined dense feature be df_j and the number of the refined dense features be N_{df} . Once the dense features are extracted,

the rest steps of the DMHLC generation is similar to the MHLC generation. Like MHLC, the f_c is computed first as:

$$f_c = \frac{1}{N_f} \sum_{i=1}^{N_f} f_i \quad \text{for } i=1,2,..N_f \quad (3.16)$$

where f_i is the spatial location of the i^{th} SIFT feature point and N_f is the total number of the SIFT feature points in the image. To find the neighbor dense features of f_i , we first compute the average of the distances between all f_i and f_c :

$$r_{mean} = \frac{1}{N_f} \sum_{i=1}^{N_f} ||f_i - f_c||_2 \quad (3.17)$$

Note that r_{mean} is used to decide the range of tattoo region in the image. All the j^{th} feature points to satisfying the equation (3.18) are then considered to be the neighbor dense features of i^{th} feature point.

$$||df_j - f_i||_2 \leq r_{th} r_{mean} \quad \text{for } j = 1, 2, ..N_{df} \quad (3.18)$$

where $r_{th}=3$ is a distance threshold to define the neighbor dense features of the SIFT feature point. We call B_i the neighbors of the i^{th} feature point and N_{B_i} the number of the neighbors. Once the neighbors of each feature point are determined, the distances and the angles between all df_j and f_i for $j \in B_i$ are computed as:

$$r_{ji} = ||df_j - f_i||_2, \quad \theta_{ji} = \arctan \frac{df_{jy} - f_{iy}}{df_{jx} - f_{ix}} \quad \text{for } j \in B_i \quad (3.19)$$

where $df_j = (df_{jx}, df_{jy})$ and $f_i = (f_{ix}, f_{iy})$. Then, we compute the polar 2D histogram, h_{m_r, n_θ}^i that represents the 2D spatial distribution of neighbors of the i^{th} feature point:

$$\begin{aligned} h_{m_r, n_\theta}^i(m, n) &= \sum_{j \in B_i} \#(r_l \leq \frac{r_{ji}}{r_{mn,i}} < r_u, \theta_l \leq \theta_{ji} - \theta'_i < \theta_u) \\ r_l &= \frac{r_{max}}{m_r}(m-1), r_u = \frac{r_{max}}{m_r}m \quad \text{for } m = 1, .., m_r \\ \theta_l &= \frac{2\pi}{n_\theta}(n-1), \theta_u = \frac{2\pi}{n_\theta}n \quad \text{for } n = 1, .., n_\theta \end{aligned} \quad (3.20)$$

where $h_{m_r, n_\theta}^i(m, n)$ is the 2D polar histogram centered on f_i , m_r is a parameter of the histogram for the number of bins for the radius, n_θ is a parameter of the histogram

for the number of bins for the angles, $r_{max}=2$ is a constant for maximum limit of a distance, θ'_i is the dominant local orientation obtained from the i^{th} SIFT feature, r_{mn_i} is the average of r_{ji} over all $j \in B_i$, and $\#()$ is a counting operator. Since we normalize r_{ji} by r_{mn_i} and align all θ_{ji} by θ'_i , the local shape context based on h_{m_r, n_θ}^i is invariant to translation, scale and rotation. We combine then multiple 2D polar histograms whose bins' sizes are changed from small to large for our local shape context to be robust to shape distortions as well as to represent detailed shape information.

$$lsc^i = \frac{1}{N_{B_i}} \begin{bmatrix} lsc_1^i \\ lsc_2^i \\ lsc_3^i \end{bmatrix} = \frac{1}{N_{B_i}} \begin{bmatrix} vec(h_{m_r, 0.5n_\theta}^i) \\ vec(h_{m_r, n_\theta}^i) \\ vec(h_{m_r, 2n_\theta}^i) \end{bmatrix} \quad (3.21)$$

where $vec()$ is the vectorization operator, lsc_1^i , lsc_2^i , and lsc_3^i are vectorized histogram whose angle parameters are $0.5n_\theta$, n_θ , and $2n_\theta$ respectively. Next, the i^{th} local shape context, lsc^i is normalized to be combined with a SIFT descriptor.

$$nlsc^i = \frac{lsc^i}{||lsc^i||_2} \quad (3.22)$$

where $nlsc^i$ is the normalized local shape context. Finally, the DMHLC descriptor for i^{th} feature point, $DMHLCD^i$ is created by :

$$DMHLCD^i = \begin{bmatrix} w_1 Sd^i \\ w_2 nlsc^i \end{bmatrix} \quad (3.23)$$

where Sd^i is the normalized i^{th} SIFT descriptor and $w_1=1$ and $w_2=128/126$ are the weights corresponding to Sd^i and $nlsc^i$. Since a SIFT descriptor and h_{m_r, n_θ}^i are invariant to translation, scale, and rotation, our $DMHLCD^i$ is also invariant to them. For the DMHLC descriptor, the same parameters as MHLC, $m_r=3$ and $n_\theta=12$ are used in this thesis.

3.3.3 Image Matching

Like MHLC descriptor, the same local feature descriptor based matching is used. Based on the local descriptor matching, the same image similarity defined in Equation(3.13)

is used. The image similarity is used to retrieve the most N similar images from the image database.

3.3.4 Experimental Results

In this experiment the two same image datasets (dataset 1 and dataset 2) in section 3.2 are used. To evaluate the performance of the DMHLC based image retrieval system we also used the Cumulative Match Characteristic (CMC) [121].

Indiana State Police Dataset (Dataset 1)

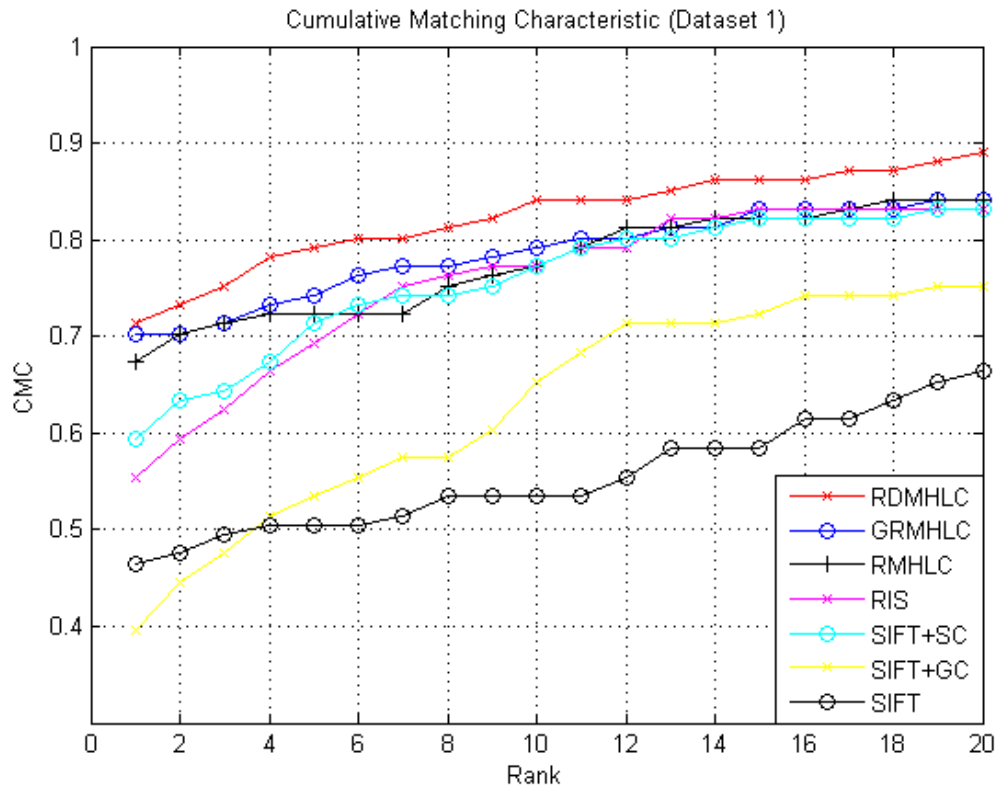


Fig. 3.10.: Retrieval performance comparison of our proposed methods, RIS [4], SIFT+SC [117], SIFT+GC [120], and SIFT [21] using CMC in Dataset 1

Table 3.5.: CMC in dataset 1 (*unit : %*)

Method	rank-1	rank-10	rank-20
RDMHLC	71.29	84.16	89.11
GRMHLC	70.3	79.21	84.16
RMHLC	67.33	77.23	84.16
RIS [4]	55.45	77.23	83.17
SIFT+SC [117]	59.41	77.23	83.17
SIFT+GC [120]	39.6	65.35	75.25
SIFT [21]	46.53	53.47	66.34

We compared our method against RIS [4], SIFT+SC [117], SIFT+GC [120], and SIFT [21] in terms of the CMC by implementing their system and testing them on dataset 1. Figure 3.6 and Table 3.2 show the experimental results. In RIS robust image similarity based on SIFT descriptor is used. SIFT+SC is the combination of SIFT and the shape context proposed in [117]. SIFT+GC is the combination of SIFT and the global context proposed in [120]. RDMHLC is the method that combines our DMHLC descriptor and the robust image similarity (RIS). RMHLC is the method that combines the MHLC descriptor and the robust image similarity (RIS). RMHLC is the method that combines the MHLC descriptor, the robust image similarity (RIS), and the global shape descriptor in Section 3.2. The experimental results demonstrate that RDMHLC achieved higher accuracy than any other methods. Even though RDMHLC does not use the global shape descriptor, it achieved better accuracy than GRMHLC. As shown in Table 3.2 RDMHLC achieves 71.29% top rank-1 accuracy and 84.16% top rank-10 accuracy in dataset 1. It outperforms RMHLC by 3.96% in top rank-1 and 6.93% in top rank-10 in dataset 1.

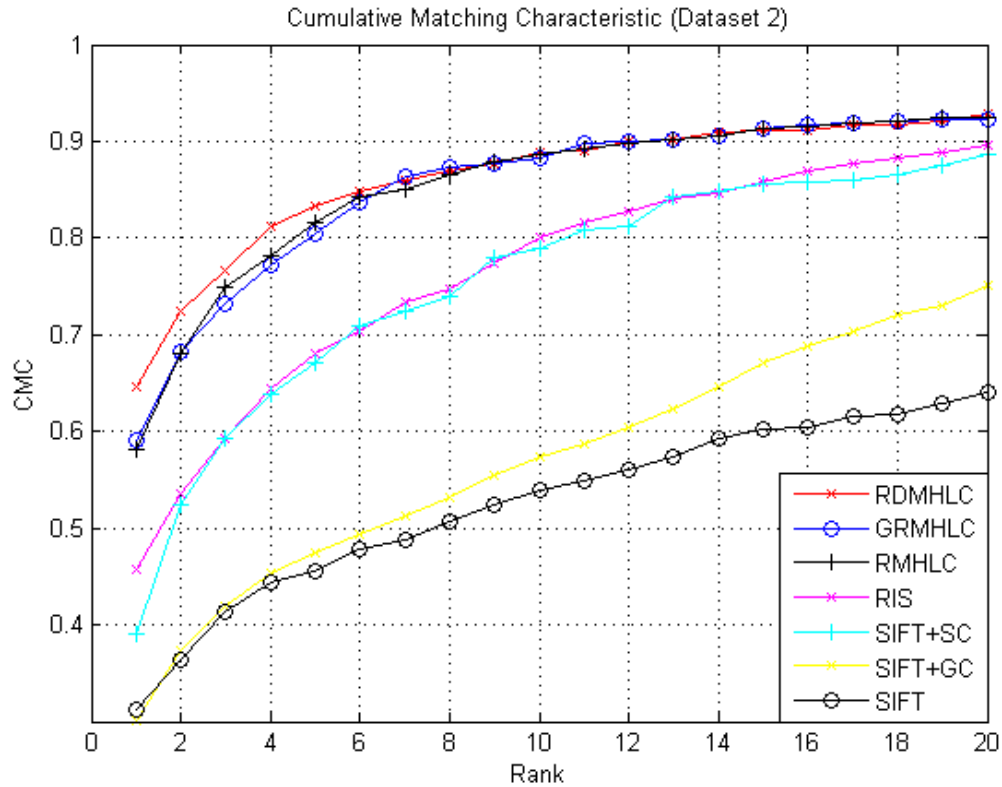


Fig. 3.11.: Retrieval performance comparison of our proposed methods, RIS [4], SIFT+SC [117], SIFT+GC [120], and SIFT [21] using CMC in dataset 2

Eviltattoo Dataset (Dataset 2)

We compared our method against RIS [4], SIFT+SC [117], SIFT+GC [120], and SIFT [21] in terms of the CMC by implementing their system and testing them on dataset 2. Figure 3.11 and Table 3.6 show the experimental results. The experimental results demonstrate that the RDMHLC achieved higher accuracy than any other methods in dataset 2 as well. As shown in Table 3.6 RDMHLC achieves 64.66% top rank-1 accuracy and 88.77% top rank-10 accuracy in dataset 2. It outperforms the RMHLC by 6.45% in top rank-1 and 0.2% in top rank-10 in dataset 2. Compared to dataset 1, RDMHLC improves the retrieval accuracy of RMHLC much more in top

Table 3.6.: CMC in dataset 2 (*unit : %*)

Method	rank-1	rank-10	rank-20
RDMHLC	64.66	88.77	92.93
GRMHLC	59.04	88.36	92.31
RMHLC	58.21	88.57	92.52
RIS [4]	45.74	80.04	89.6
SIFT+SC [117]	39.09	79.00	88.57
SIFT+GC [120]	30.15	57.38	75.05
SIFT [21]	31.19	53.85	64.03

rank-1 than top rank-10 in dataset 2. It means that DMHLC can describe the tattoo more accurate than MHLC.

3.4 Tattoo Image Retrieval System Based On Inductive Matching

Many image retrieval methods based on image matching [6, 9, 31, 122–126] have used the pairwise image similarity between an input image and a database image to retrieve the most similar images to the input image. However, the pairwise image similarity based image retrieval system often show wrong image retrieval results when there are the variations of appearances and shapes between the same class images. In this case, the image similarities between all the database images can be additionally considered to improve the image retrieval accuracy. For example, assume that we have three images, A, B, and C, and each of them includes the rabbit with different appearance: C has partially similar appearance with each of A and B, but A looks very different B. We put B and C in the image database that includes lots of images, and we input A to retrieve the images B and C. A is compared with all the database images using a certain pairwise similarity. According to the image similarities, A is similar to C, B is similar to C, but A is not similar to B. In this case, the pairwise image similarity

between two images is not enough to retrieve a correct images from database. To solve this problem, the pairwise similarity between A and C and the pairwise similarity between B and C can be additionally used to compute the similarity between A and B. The similarity relations of the database images have been used in the information retrieval as well as the image retrieval system [127–139]. Most of the retrieval systems [127–137] are based on a diffusion process. The diffusion process computes the affinity (or similarity) matrix of the images. Note that the $(i, j)^{th}$ element of the affinity matrix is the pairwise image similarity between the i^{th} image and j^{th} image. Then, the affinity matrix is iteratively updated using a random walk technique, and the updated affinity matrix generates the new image similarity that incorporate the similarity relations of the images. In [138, 139] an inductive matching technique was used. In the inductive matching the similarity relations of the images are used simply. For example, given an input image, the most similar M_1 images are retrieved from a database using the pairwise image similarities between the input image and all the database images. Then, the pairwise image similarities between the M_1 images and all the database images are computed again, and these are added or multiplied to the previous pairwise image similarities. In both the diffusion process and the inductive matching, there is one important assumption: the most images that are distributed closely in the image feature space should belong to the same class. Note that the shorter distance between two image features is, the more similar the two images are. The Figure 3.12 shows the example of it. As depicted in 3.12b, the diffusion process can cluster the database images into the correct classes. However, if the images that belong to the same class are not distributed closely, the diffusion process or the inductive matching can cause worse results. Especially, when there are much more images that are not related to an input images than the related images in the databases, these methods make worse results. In this section, therefore, we introduce the modified inductive matching that even works well in the case. Different from the inductive matching in [139], the modified inductive matching retrieve the most dissimilar M_2 database images to an input image first. Then, when the similarity

between the input image and one database image is computed, the mean of the image similarities between the M_2 database images and the database images is subtracted from the pairwise similarity between the input image and the database image.

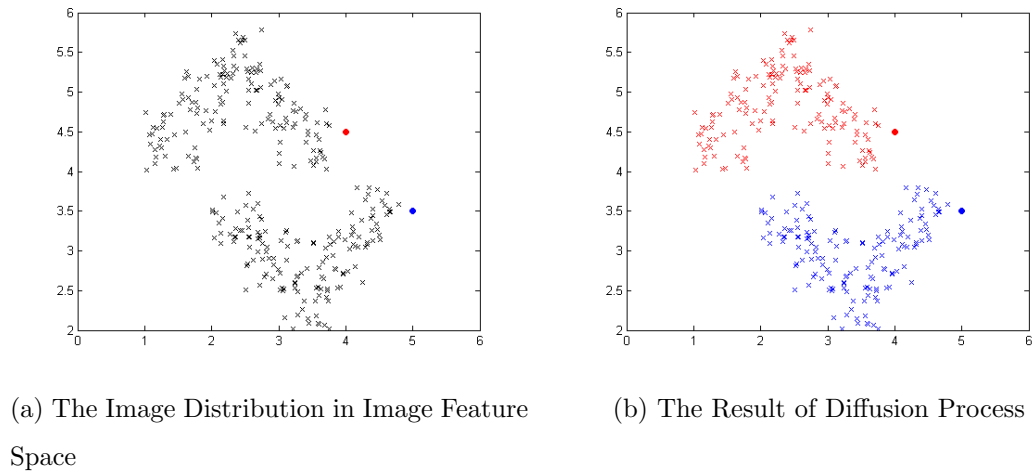


Fig. 3.12.: Example of the diffusion process: In 3.12a the black dots are the database images that belong to two different classes and blue and red dots are input images. The diffusion process can cluster them into two class correctly

3.4.1 System Overview

Figure 3.13 shows the block diagram of tattoo image retrieval system based on the inductive matching. The system is divided into two parts. In the archiving process, we first generate an image descriptor from each image in our database. Note that any image descriptor can be used here. Using these image descriptors we generate affinity matrix that consists of all the pairwise image similarities of the database images. Our database then consists of images, image descriptors, and the affinity matrix. Given an input image that we would like to compare to our database for similar images, we construct an image descriptor. We then compare the image descriptor from the input image against the image descriptors in our database using the modified inductive

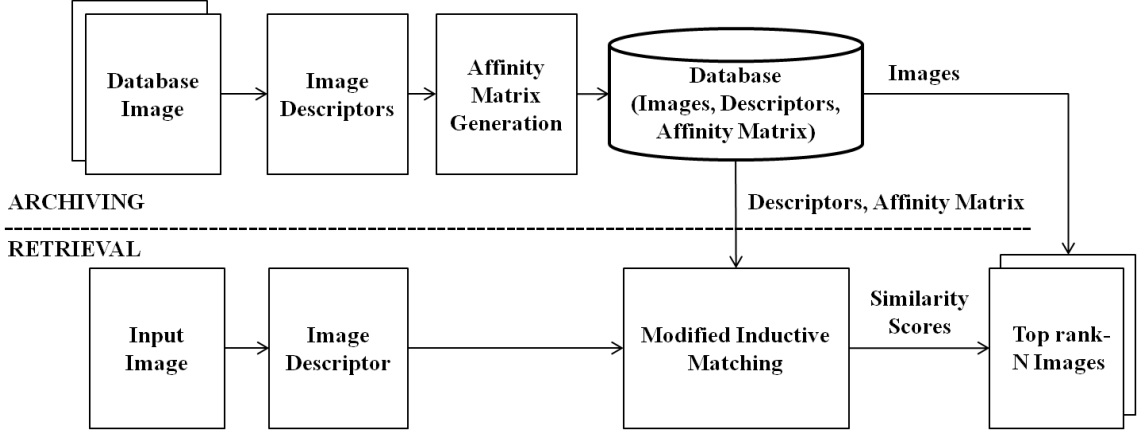


Fig. 3.13.: Tattoo image retrieval system based on Modified Inductive Matching

matching method. It returns image similarity scores between an input image and a database image to retrieve the top N similar images.

3.4.2 The Modified Inductive Matching

Inductive matching is the one of the methods that re-rank the retrieved images [139]. In the inductive matching, the most similar M_1 database images to an input image are additionally used to define each image similarity between the input image and each of database images. For this, the pairwise image similarity based an image descriptor should be very accurate to cluster the same class images into one group. Since, however, there are lots of variations can exist in the same class images, it is not easy to decide that all of them are similar using the pairwise image similarity. Especially, when there are lots of multi class images in the image database, some of the same class images can be easily considered to be similar to different class images. For the case, we modify the inductive matching The modified inductive matching retrieve the most dissimilar M_2 database images to the input image, I_q , first. Then, when the similarity between the input image and the k^{th} database image is computed, the mean of the image similarities between the M_2 database images and the k^{th} database image is subtracted from the pairwise similarity between the input image and the

k^{th} database image. In tattoo image retrieval with inductive matching, we use the robust image similarity defined in Equation(3.13) based on DMHLC descriptors for the pairwise image similarity between two images. Before inductive matching, the robust image similarity should be first normalized as:

$$S_{norm}(I_q, I_k) = \frac{1}{N_f} \sum_{i=1}^{N_f} x_i \left(\frac{1}{m^i(I_k)} \log \frac{N_G}{n^i} \right) \quad (3.24)$$

where I_q is an input image, I_k is the k^{th} database image, N_f is the number of the SIFT features extracted from I_q , S_{norm} is the normalized robust image similarity between I_q and I_k , $m^i(I_k)$ is the number of feature points including the i^{th} feature point in I_q that are matched to the same feature point in I_k , n^i is the number of database images that have feature points matched to i^{th} feature point in I_q , and N_G is the total number of database images. x_i is 1 if the ratio of the closest match to the second closest match between $DMHLC D^i$ in I_q and all $DMHLC D^j$ in I_k is less than pre-defined threshold.

Our new image similarity based on inductive matching between I_q and I_k is then defined as:

$$S_{idt}(I_q, I_k) = S_{norm}(I_q, I_k) - \frac{1}{M_2} \sum_j^{M_2} S_{norm}(I_{DS^j}, I_k) \quad (3.25)$$

where $S_{idt}(I_q, I_k)$ is the inductive matching image similarity between I_q and I_k , $S(I_q, I_k)$ is the normalized pairwise similarity between I_q and I_k , and I_{DS^j} is the j^{th} image in the most dissimilar M^2 retrieved images retrieved to I_q from the database. The new similarity based on inductive matching is used to retrieve the most similar N images from the image database.

3.4.3 Experimental Results

In this experiment the eviltattoo dataset (dataset 2) is used. To show that our modified inductive matching can be used with any kinds of image descriptor, the MHLC descriptor as well as the DMHLC descriptor is used as well. Also, to show that any kinds of image similarity can be used with our inductive matching, the

image similarity used in [21] as well as the normalized robust image similarity in Equation 3.24. Therefore, the 4 different pairwise image similarities are computed here, and our modified inductive method is combined with the each of them. The inductive matching [139] and Label Propagation [134] are also combined with each of the pairwise similarities to compare against the modified inductive matching. Note that the Label Propagation is one of the method that uses diffusion process. To evaluate the performance of the image retrieval based on the inducve matching we also used the Cumulative Match Characteristic (CMC) [121].

DMHLC Descriptors With Robust Image Similarity

Table 3.7.: CMC using DMHLC with robust image similarity (*unit : %*)

Method	rank-1	rank-10	rank-20
Modified Inductive Matching	67.78	89.81	93.14
Inductive Matching [139]	59.67	86.90	92.10
Label Propagation [134]	60.91	86.90	93.56
Pairwise	64.66	88.77	92.93

Figure 3.14 and Table 3.7 show the experimental results when the DHMLC descriptors are used with the robust image similarity. The experimental results demonstrate that the modified inductive matching achieved higher accuracy than any other methods in the most top ranks. As shown in Table 3.7 the modified inductive matching achieves 67.78% top rank-1 accuracy and 89.81% top rank-10 accuracy. It improves the pairwise image similarity by 3.12% in top rank-1 and 1.04% in top rank-10. However, the inductive matching achieves worse results in all the ranks than the pairwise image similarity, and the Label Propagation achieves worse results in from top 1 rank to top 13 rank. That is because the the same class database images in dataset 2 are not clustered well using the pairwise image similarity.

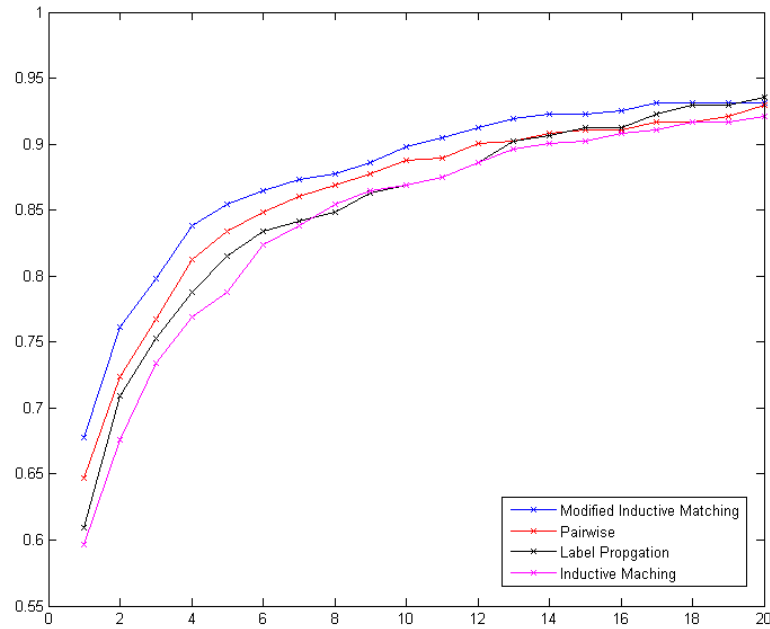


Fig. 3.14.: CMC curve using DMHLC with robust image similarity

DMHLC Descriptors With Different Image Similarity

Table 3.8.: CMC using DMHLC With image similarity in [21] (*unit : %*)

Method	rank-1	rank-10	rank-20
Modified Inductive Matching	55.09	86.49	92.31
Inductive Matching [139]	35.76	64.45	81.29
Label Propagation [134]	29.11	73.60	84.82
Pairwise	42.62	76.92	89.40

Figure 3.15 and Table 3.8 show the experimental results when the DHMLC descriptors are used with the image similarity in [21]. The experimental results demonstrate that the modified inductive matching achieved higher accuracy than any other methods. As shown in Table 3.8 the modified inductive matching achieves 55.09% top

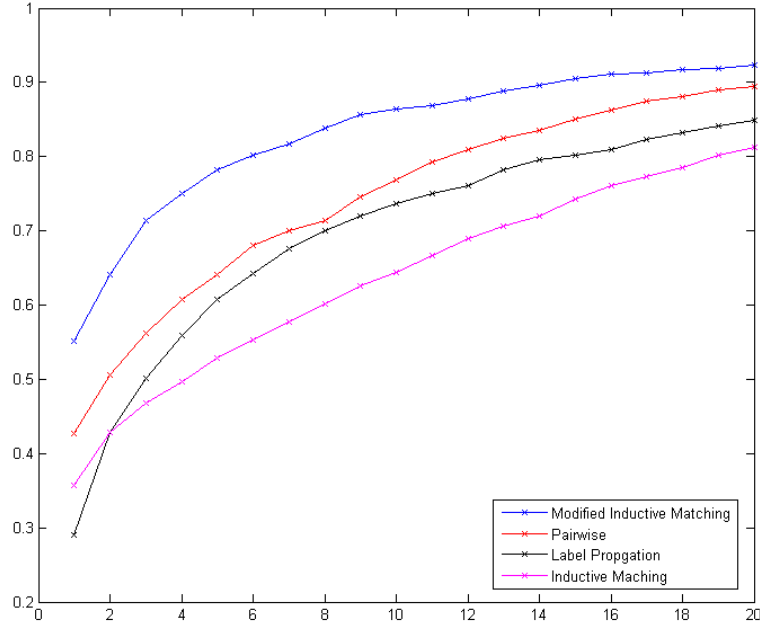


Fig. 3.15.: CMC curve using DMHLC with image similarity in [21]

rank-1 accuracy and 86.49% top rank-10 accuracy. It improves the pairwise image similarity by 12.47% in top rank-1 and 12.57% in top rank-10. However, both the inductive matching and Label Propagation achieves worse results in all the ranks than the pairwise image similarity. That is because the the same class database images in dataset 2 are not clustered well using the pairwise image similarity.

MHLC Descriptors With Robust Image Similarity

Figure 3.16 and Table 3.9 show the experimental results when the HMLC descriptors are used with the robust image similarity. The experimental results demonstrate that the modified inductive matching achieved higher accuracy than any other methods. As shown in Table 3.9 the modified inductive matching achieves 61.12% top rank-1 accuracy and 88.98% top rank-10 accuracy. It improves the pairwise image similarity by 2.91% in top rank-1 and 0.41% in top rank-10. However, both the in-

Table 3.9.: CMC using MHLC With robust image similarity (*unit : %*)

Method	rank-1	rank-10	rank-20
Modified Inductive Matching	61.12	88.98	94.18
Inductive Matching [139]	53.43	86.07	90.85
Label Propagation [134]	51.14	85.86	91.89
Pairwise	58.21	88.57	92.52

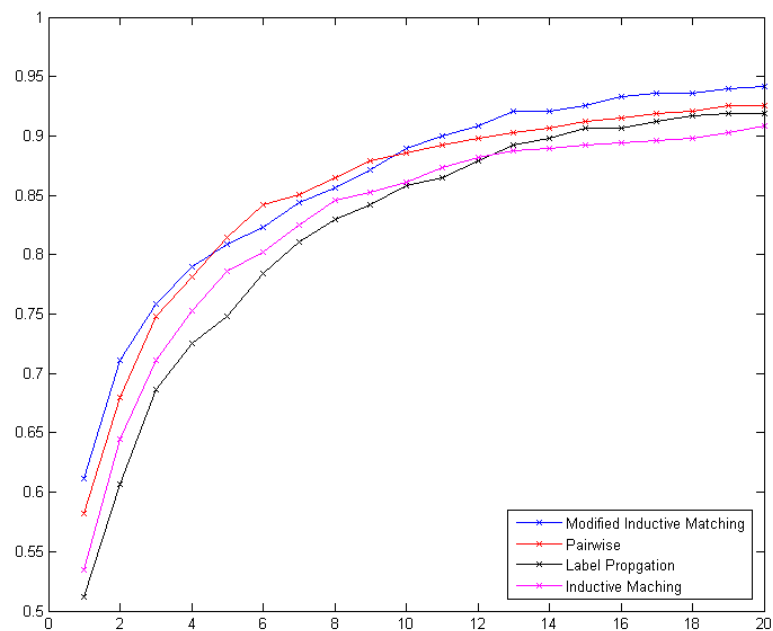


Fig. 3.16.: CMC curve using MHLC with robust image similarity

ductive matching and Label Propagation achieves worse results in all the ranks than the pairwise image similarity. That is because the the same class database images in dataset 2 are not clustered well using the pairwise image similarity.

Table 3.10.: CMC using MHLC with image similarity in [21] (*unit : %*)

Method	rank-1	rank-10	rank-20
Modified Inductive Matching	51.77	84.62	93.35
Inductive Matching [139]	39.92	70.27	81.08
Label Propagation [134]	25.57	69.85	85.86
Pairwise	44.07	78.79	89.19

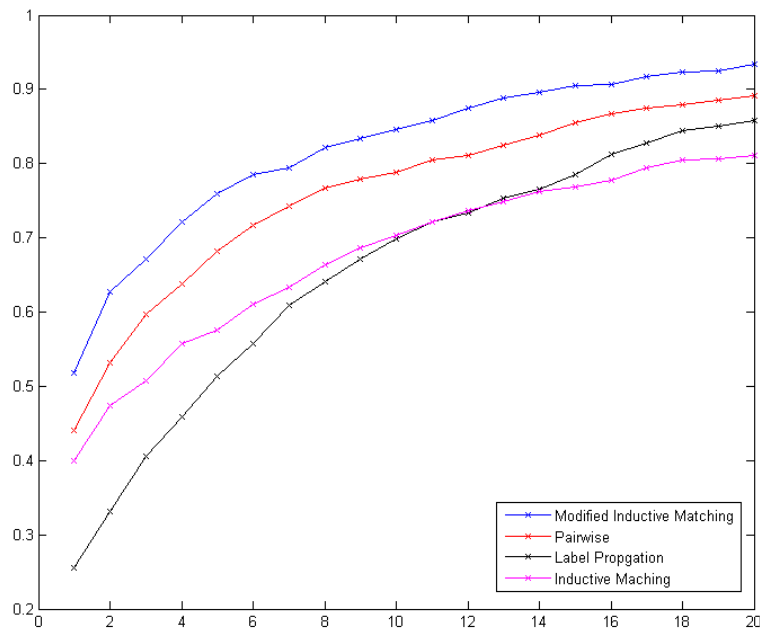


Fig. 3.17.: CMC curve using MHLC with image similarity in [21]

MHLC Descriptors With Different Image Similarity

Figure 3.17 and Table 3.10 show the experimental results when the MHLC descriptors are used with the image similarity in [21]. The experimental results demonstrate that the modified inductive matching achieved higher accuracy than any other methods. As shown in Table 3.10 the modified inductive matching achieves 51.77% top

rank-1 accuracy and 88.98% top rank-10 accuracy. It improves the pairwise image similarity by 7.7% in top rank-1 and 5.83% in top rank-10. However, both the inductive matching and Label Propagation achieves worse results in all the ranks than the pairwise image similarity. That is because the the same class database images in dataset 2 are not clustered well using the pairwise image similarity.

3.5 Tattoo Image Retrieval For Tattoo Identification

NIST (National Institute of Standards and Technology) conducted the Tattoo Recognition Technology - Challenge (Tatt-C) in early 2015 for developing tattoo image recognition methods. The contest used a tattoo image dataset (tatt-c database) provided by NIST and the FBI for doing its evaluations [13]. The tatt-c database consists of five different datasets focused on five primary use cases: tattoo identification, region of interest, mixed media, tattoo similarity, and tattoo detection [13]. The goal of the tattoo identification challenge was to develop image matching methods that allow one tattoo image to be matched to the tattoo image taken from the same subject at different time.

In this section we describe our submission to the Tatt-C tattoo identification challenge. The image retrieval system for this challenge is almost same as the system described in Section 3.2. The differences are that the global shape descriptor and the global shape matching are not used here because the tattoo identification dataset has non-cropped images that include background. Also, the computation of normalization factor used in Equation(3.3) is different. Last, SIFT descriptor is used as our image descriptor instead of MHLC descriptor in image matching when the number of extracted SIFT features is not enough. Note that our experimental results for TID were reported by NIST in [11].

3.5.1 System Overview

There are three differences between this system and the system in Section 3.2. Since all the images in the TID dataset are not cropped, the global shape descriptor and global shape matching described in Section 3.2 cannot be used here. Also, the normalization factor, r_{mn_c} used in Equation(3.3) is computed based on SIFT feature points filtered out from visual saliency map by estimating the scale of a tattoo object using the SIFT feature points on a tattoo region. The MHLC descriptor is not accurate when the number of the extracted SIFT feature points is not enough for shape representation. In that case SIFT is used as our image descriptor instead in image matching. Since the MHLC consists of SIFT and the combined multiple histogram vectors, SIFT descriptor is obtained from the MHLC descriptor by choosing first 128 elements of the MHLC descriptor. Except these modifications, others are the same in the system in Section 3.2.

Before description of the system, we define three words here: probe image, gallery image and background image. Probe image is an input image that is presented to our image retrieval system for TID matching. Gallery images (or database images) are images in our database, and the most similar N images to a probe image are retrieved from the gallery images. Background image is an image that is added to the gallery images as “distractor” to confuse the retrieve process. Since these words are defined in the NIST tattoo recognition challenge, we used these words in Section 3.5 and Section 3.6 instead of an input image and a database image.

Figure 3.18 shows the block diagram of our tattoo image retrieval system for tattoo identification (TID). For each gallery image in our database we obtain the MHLC descriptors that are associated with the image. In the retrieval process when a probe image is presented to this system, we compute the MHLC descriptors. We then compare the MHLC descriptors from the probe image against the gallery images in our database using the matching method based on the robust similarity. Note that if the number of the extracted SIFT feature points in the probe image is smaller than

a pre-defined threshold, SIFT descriptors are obtained from the MHLC descriptors of all the gallery images and the probe image. Then, they are compared using the matching method based on the robust similarity. The matching method returns a score to retrieve the top N matched images.

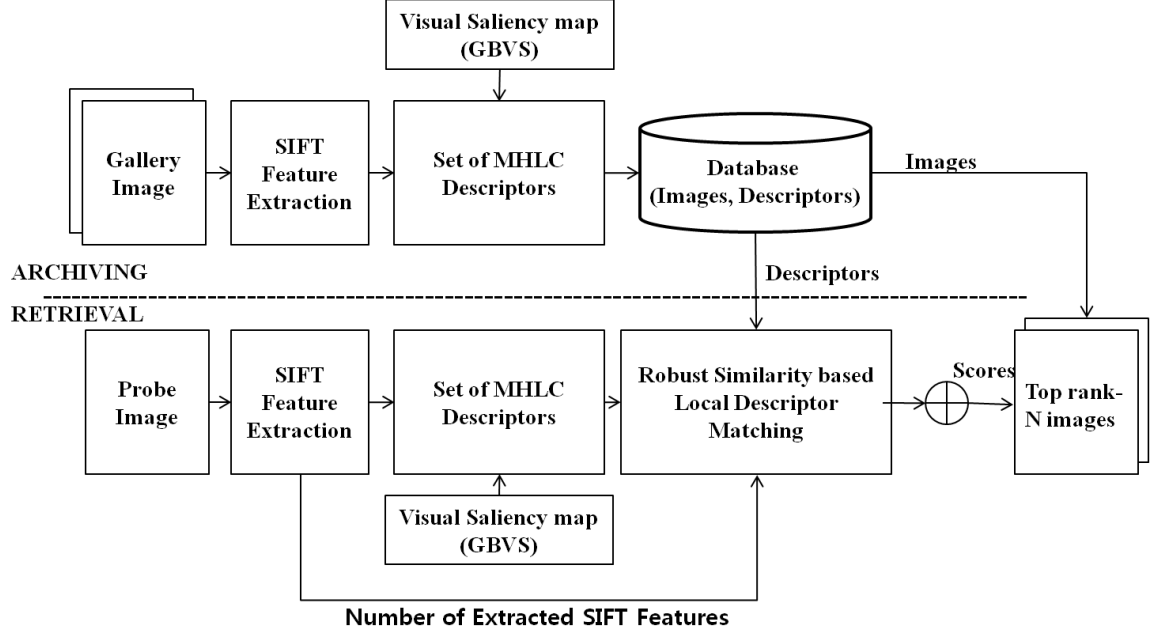


Fig. 3.18.: Tattoo image retrieval system for tattoo identification

3.5.2 MHLC Descriptor

Since the computation of normalization factor, $r_{mn,c}$ used in Equation(3.3) is different from Section 3.2, we will describe this only here. The $r_{mn,c}$ is computed here as:

$$r_{mn,c} = \frac{1}{N_s} \sum_{i=1}^{N_s} \|f_{is} - f_{cs}\|_2 \quad (3.26)$$

Table 3.11.: The description for our datasets

	Number of Probe Image	Number of Gallery Image	Number of Background Image	Number of Different Tattoo Object	Cropped
Dataset 4	157	215	4332	157	No

where f_{is} is the spatial location of the i^{th} SIFT feature point within region R_s , f_{cs} is the average of f_{is} , R_s is the salient region detected using GBVS saliency map, N_s is the total number of SIFT feature points in the R_s

Other process to compute the MHLC descriptor is the same as in Section 3.2.

3.5.3 Image Matching

The same image matching based on Equation(3.13) is used. Only difference is that SIFT is used as our image descriptor when the number of the extracted SIFT feature points in the probe image is smaller than a pre-defined threshold. Since the MHLC consists of SIFT and the combined multiple histogram vectors, SIFT descriptor is obtained from the MHLC descriptor by choosing first 128 elements of the MHLC descriptor.

3.5.4 Experimental Results

In this experiment, tattoo identification dataset (dataset 4) is used to evaluate our tattoo image retrieval system. This dataset is obtained from NIST tattoo challenge dataset [13]. The NIST tattoo identification dataset consists of 4704 tattoo images: 157 probe images, 215 gallery images and 4332 background images [13]. Note that the background images are added to the gallery images in this dataset as “distractors”. The description of our datasets is summarized in Table 3.11. To evaluate the

Table 3.12.: CMC in dataset 4 (*unit : %*)

	With Background Image			Without Background Image		
Method	rank-1	rank-10	rank-20	rank-1	rank-10	rank-20
Proposed	96.15	98.71	98.71	99.38	99.38	100
[4]	95.52	96.81	97.48	96.81	100	100

performance of our proposed method we used the Cumulative Match Characteristic (CMC) [121] to obtain the top-N (N=20) rank retrievals from our database.

NIST TID Dataset (Dataset 4)

For the dataset 4, the image retrieval system illustrated in Section 3.5 is used. The TID dataset is split into 5 sub-datasets for 5-fold cross validation. The CMC score is then computed for each sub-dataset and the average CMC score is obtained. We compared our method against the method described in [4] in terms of the average CMC.

We shall refer to method described in [4] as the MSU method here. Figure 3.19, Figure 3.20, and Table 3.12 show the experimental results. Note that the dataset used for Figure 3.19 includes background (or distractor) images, but the dataset used for Figure 3.20 does not include the background (or distractor) images. Our experimental results show that our method is more accurate than MSU in all ranks for dataset 4 with background images. Our method achieves 96.15% top rank-1 accuracy and 98.71% top rank-10 accuracy in dataset 4 with background and 99.38% top rank-1 accuracy and 99.38% top rank-10 accuracy in dataset 4 without background. It outperforms MSU by 0.6% top rank-1 with background images and 2.5% top rank-1 without background images. As shown in 3.13 our method (Purdue phase 2) also outperforms 5 different methods reported in the NIST challenge [11].

Table 3.13.: CMC and MAP in TID dataset with background images as reported in [11]

Method	Submission	CMC				MAP
		Rank 1	Rank 10	Rank 100	Rank 300	
CEA1	phase 1	0.898	0.898	0.904	0.924	0.874
CEA2	phase 2	0.962	0.962	0.962	0.975	0.937
Compass	phase 2	0.089	0.140	0.586	0.650	0.121
FraunhoferIOSB	phase 1	0.854	0.854	NA	NA	0.842
FraunhoferIOSB	phase 2	0.968	0.968	0.968	0.975	0.954
MITRE	phase 2	0.529	0.879	0.943	0.975	0.676
Morpho Track	phase 1	0.987	0.987	0.994	0.994	0.988
Morpho Track	phase 2	0.994	0.994	1.000	1.000	0.994
Purdue	phase 1	0.968	0.981	0.994	0.994	0.967
Purdue	phase 2	0.962	0.987	0.994	0.994	0.964

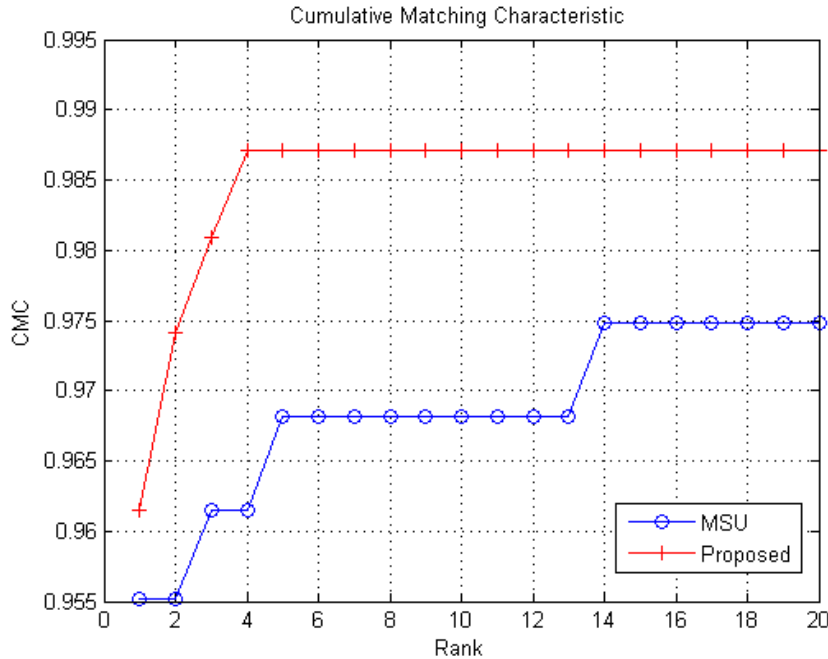


Fig. 3.19.: Retrieval performance of our proposed method and MSU [4] using CMC in TID dataset with background image

3.6 Tattoo Image Retrieval For Region of Interest

In early 2015, NIST (National Institute of Standards and Technology) conducted the Tattoo Recognition Technology - Challenge (Tatt-C) for developing tattoo image recognition methods [12]. The challenge used a tattoo image dataset (tatt-c database) provided by NIST and the FBI for doing the evaluations [13]. The Tatt-C database consists of five datasets focused on five primary use cases: tattoo identification, region of interest, mixed media, tattoo similarity, and tattoo detection [12, 13]. The goal of the region of interest challenge was to develop image matching methods that allowed subregions of interest from one tattoo image to be matched to another tattoo image. Note that the “other tattoo” image is taken from the same subject of the original image that includes the subregions of interest but at a different time. Therefore, there are illumination changes and some distortions between the images.

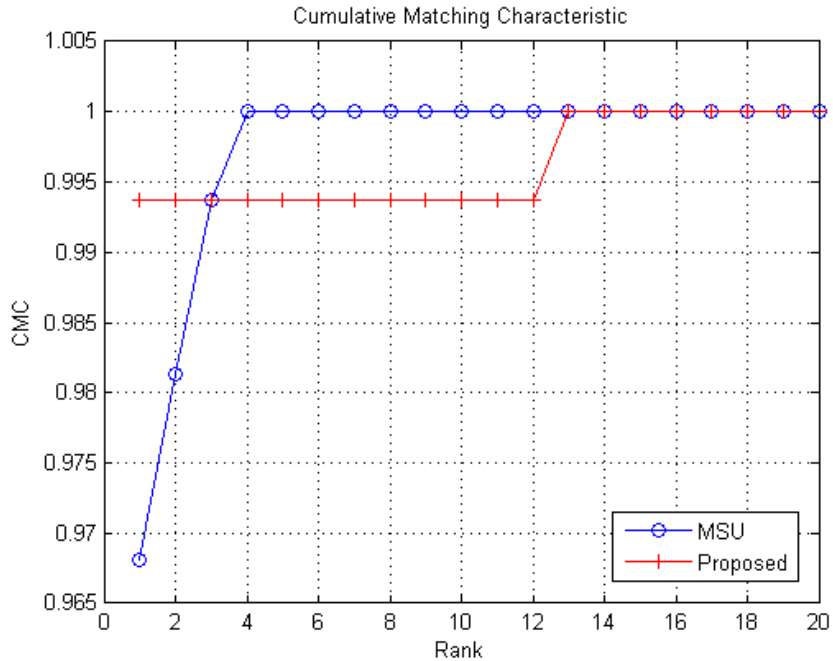


Fig. 3.20.: Retrieval performance of our proposed method and MSU [4] using CMC in TID dataset without background image

In this section we describe our submission to the Tatt-C region of interest challenge [140]. Our contribution is an image descriptor based on local self similarity [62] and SIFT [21] that can represent a subregion of a tattoo image. We introduce a weighted distance similarity metric to retrieve the most similar images from the test dataset. Note that our experimental results were reported by NIST along with the results of other participants of the Tatt-C Challenge in [11]. However, this report [11] does not provide any technical description of our methods which is the basis of this method.

3.6.1 System Overview

Figure 3.21 shows a block diagram of our proposed tattoo image retrieval system for region of interest (ROI). For each image in our database we obtain two descrip-

tors that are associated with the image and are used for ROI matching. These two descriptors are Local Self Similarity (LSS), described below, and SIFT. Images in our database are sometimes referred to as gallery images.

In the retrieval process when a new image (also known as the probe image) is presented to our system for ROI matching we first compute the SIFT and LSS descriptors. We then compare the image descriptors from the probe image against the gallery images in our database using the matching method based on our proposed robust weighted distance similarity. The matching method returns a score to retrieve the top N matched images.

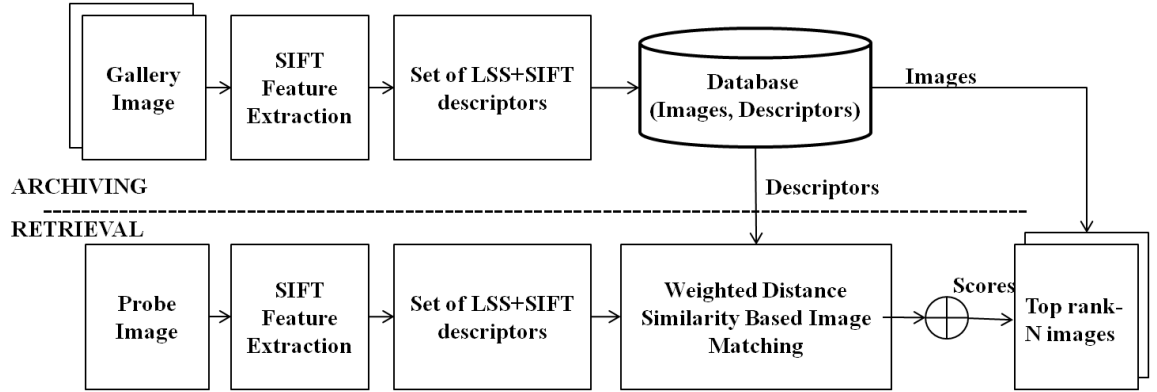


Fig. 3.21.: Tattoo image retrieval system for region of interest

3.6.2 Scale Invariant Feature Transform

SIFT (Scale Invariant Feature Transform) has been widely used in object recognition since it is invariant to scale and rotation and are robust to affine deformation, illumination change, and change in 3D viewpoint [21]. Since SIFT has proved to be effective in other tattoo image retrieval applications [3, 4, 29], we also used SIFT. Each SIFT feature point is identified as a local minima and maxima of the Difference of Gaussians (DoG) images across scales [21]. To obtain stable features, we reject the feature points with low contrast. If the absolute value of the DOG for a feature

point is less than the peak threshold $t_p = 1.5$, we discard the feature. The SIFT descriptor is then generated from the feature point by computing the orientation of the gradient vector for each pixel in the feature point's neighborhood and finding a normalized histogram of gradient orientations. Each SIFT descriptor is represented as a 128 dimensional vector.

3.6.3 Local Self Similarity

In [62] the Local Self Similarity (LSS) image descriptor that uses the similar color of neighbor pixels was described. Since the same class objects have similar parts of the objects that have similar color, the local self similarity is an important features that can describe an image object. This descriptor can also represent the local region of object well when there are small object distortions. For this reason, the self similarity feature has been widely used in image classification and image matching. [65,141–143] We also use LSS to generate our image descriptor. An example of the LSS descriptor is shown in Figure 3.22. The i^{th} SIFT feature f_i is first determined. Then a 5×5 RGB color image patch centered on f_i (red box) is compared with a 5×5 color image patch (green box) by moving the patch within a larger region (41×41) centered on f_i (blue box). $SSD_{f_i}(x, y)$ is the sum of square differences (SSD) between the RGB color patches:

$$SSD_{f_i}(x, y) = \sum_{k \in \{R, G, B\}} \sum_{m=1}^5 \sum_{n=1}^5 (r_{mnk} - g_{mnk})^2 \quad (3.27)$$

where (x, y) is the coordinate of the center of the green box, r_{mnk} and g_{mnk} are pixels in red box and green box in each. Then the “correlation surface,” $S_{f_i}(x, y)$, is obtained from $SSD_{f_i}(x, y)$ as

$$S_{f_i}(x, y) = \exp\left(-\frac{SSD_{f_i}(x, y)}{C_n}\right) \quad (3.28)$$

where $C_n = 75$ is a constant for normalization. Note that $S_{f_i}(x, y)$ represents how much the patch centered on (x, y) is correlated with the patch centered on f_i .

The maximal value of $S_{f_i}(x, y)$ within each bin of the 2D polar histogram, shown in Figure 3.22, (12 bins for the angle, 4 bins for the radius) is computed. The 2D polar histogram whose bins filled with the maximal value is vectorized to generate lss_{f_i} which is the LSS descriptor for f_i . Next, lss_{f_i} is normalized by:

$$nlss_{f_i} = \frac{lss_{f_i}}{\|lss_{f_i}\|} \quad (3.29)$$

where $nlss_{f_i}$ is the normalized LSS descriptor.

3.6.4 Image Descriptor

A subregion of a tattoo image has less discriminative power than an entire tattoo image because the subregion has less information relative to the tattoo object. Therefore, an image descriptor that represents a subregion should have discriminable power to find a subregion for matching. We combine two image descriptors: SIFT and LSS described from Sections 3.6.2 and 3.6.3. A histogram of pixel gradients is used for the SIFT descriptor and a similar color pattern is used for the LSS descriptor. Each descriptor uses different types of image properties which allows for better discrimination of the image sub-region.

Finally, the image descriptor, ld_{f_i} is constructed by :

$$ld_{f_i} = \begin{bmatrix} w_1 nlss_{f_i} \\ w_2 nsd_{f_i} \end{bmatrix} \quad (3.30)$$

where nsd_{f_i} is the normalized SIFT descriptor for f_i and w_1 and w_2 are the weights corresponding to $nlss_{f_i}$ and nsd_{f_i}

3.6.5 Weighted Distance Similarity For Image Matching

Since image matching uses the descriptors, an appropriate image similarity metric is important for image retrieval. The similarity metric in [4] uses the relationship

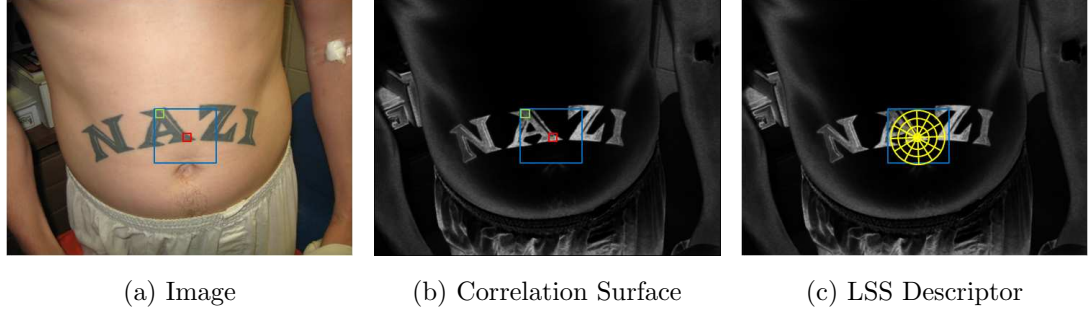


Fig. 3.22.: LSS descriptor

between the matched image features, but it does not include the distance between the feature descriptors. Therefore, our weighted distance similarity (WDS) uses the distances of the feature descriptors. Our image similarity is defined as :

$$S_{WDS}(I_p, I_k) = \sum_{i=1}^{N_f} x_i \left(\frac{1}{m^i(I_k)} \log \frac{N_G}{n^i} \right) \left(\frac{1}{1 + e^{d_1^i}} \right) \quad (3.31)$$

where S_{WDS} is the weighted distance similarity metric, I_p is a probe image, I_k is the k^{th} gallery image, $m^i(I_k)$ is the number of feature points including the i^{th} feature point in I_p that are matched to the same feature point in I_k , n^i is the number of database images that have feature points matched to i^{th} feature point in I_p , N_G is the total number of gallery images, and N_f is the number of SIFT features in I_p . x_i is 1 if the ratio of the closest distance (d_1^i) to the second closest distance (d_2^i) between ld_{f_i} in I_p and all ld_{f_j} in I_k is less than pre-defined threshold, $T_{tr} = 0.6$. Otherwise, x_i is 0. This is different from the similarity metric in [4]. Our S_{WDS} includes d_1^i combined with a sigmoid function. For ROI image retrieval, we choose the most similar tattoo images in the database to maximize the S_{WDS} .

3.6.6 Experimental Results

In this experiment we use three different metrics to evaluate the performance of our proposed method on the NIST region of interest (ROI) Tatt-C citeNgan2015

Table 3.14.: The description for our datasets

	Number of Probe Image	Number of Gallery Image	Number of Background Image	Number of Different Tattoo Object	Cropped
Dataset 5	297	157	4332	157	No

dataset (dataset 5) : the Cumulative Match Characteristic (CMC) [121], precision and recall [144], and Mean Average Precision (MAP) [145]. First, the CMC based on top-20 rank retrievals from our database is used. The CMC score at the top N rank is defined as the ratio of the number of the correctly matched probe images within the top N ranks to the total number of probe images. In this metric we want to achieve a higher CMC score at ranks with lower N . The NIST Tatt-C region of interest dataset consists of 4786 tattoo images: 297 probe images, 157 gallery images and 4332 background images [13]. Note that background images are added to the gallery images as “distractors” to confuse the retrieve process. The description of our datasets is summarized in Table 3.14.

NIST ROI Dataset (Dataset 5)

The ROI dataset is split into 5 sub-datasets for 5-fold cross validation. The CMC score is then computed for each sub-dataset and the average CMC score is obtained. We compared our method against the method described in [4] in terms of the average CMC. We shall refer to this method [4] as the “MSU method”. Figure 3.23, Figure 3.24, Table 3.15, and Table 3.16 show the CMC results. Note that the dataset used for Figure 3.23 and Table 3.16 includes distractor images, but the dataset used for Figure 3.24 and Table 3.15 does not include the distractor images. We used our image descriptors with the similarity metric described in [4] for “Proposed (Without WDS)” and compared it with “WDS” described by Equation 3.31 using our image

descriptor as shown in Figure 3.23, Figure 3.24, Figure 3.25. Our experimental results show that our method without WDS is more accurate than MSU in ranks with lower N while MSU's method is more accurate than our method without WDS in ranks with higher N . However, our method with WDS outperforms the MSU method in all ranks. The method with WDS improves 2.35% more than the MSU method for top rank-1 accuracy by achieving 96.61% (MSU : 94.26% for top rank-1) on the ROI dataset without background. The method with WDS improves 2.7% more than the MSU method for top rank-1 accuracy by achieving 91.56% (MSU : 88.86% for top rank-1) on the ROI dataset with background.

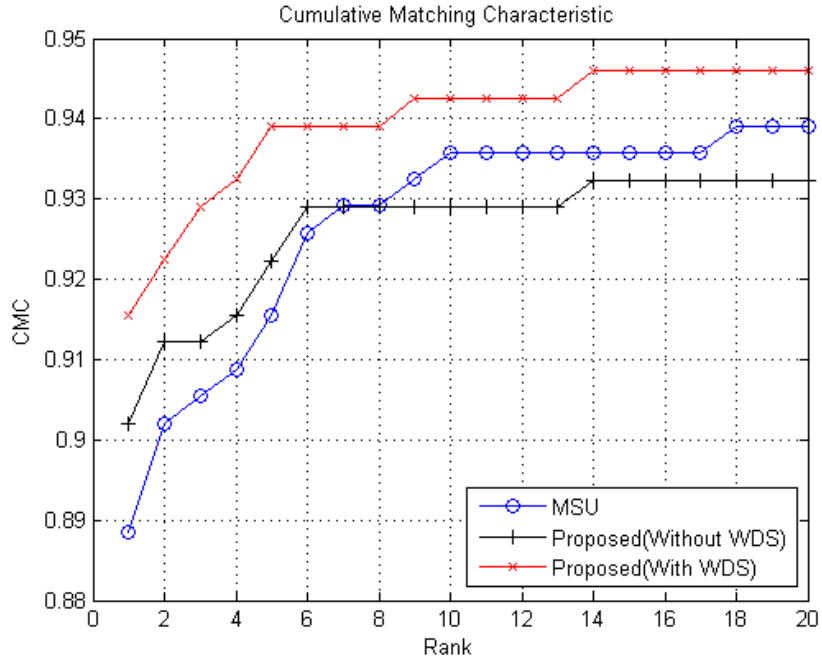


Fig. 3.23.: Retrieval performance of our proposed methods and the MSU method using CMC in the ROI dataset with background image

We also used precision and recall to compare our method with the method to the MSU method [4]. Precision and recall are defined as (using the NIST definition in [11]):

Table 3.15.: CMC and MAP for the ROI dataset without background images

Method	CMC					MAP
	Rank 1	Rank 5	Rank 10	Rank 15	Rank 20	
MSU	0.9426	0.9626	0.9830	0.9830	0.9865	0.9538
Proposed (Without WDS)	0.9423	0.9763	0.9830	0.9830	0.9896	0.9574
Proposed (With WDS)	0.9661	0.9796	0.9865	0.9898	0.9932	0.9720

Table 3.16.: CMC and MAP in ROI dataset with background image

Method	CMC					MAP
	Rank 1	Rank 5	Rank 10	Rank 15	Rank 20	
MSU	0.8886	0.9156	0.9358	0.9358	0.9391	0.9015
Proposed (Without WDS)	0.9021	0.9223	0.9289	0.9323	0.9323	0.9068
Proposed (With WDS)	0.9156	0.9391	0.9425	0.9460	0.9460	0.9240

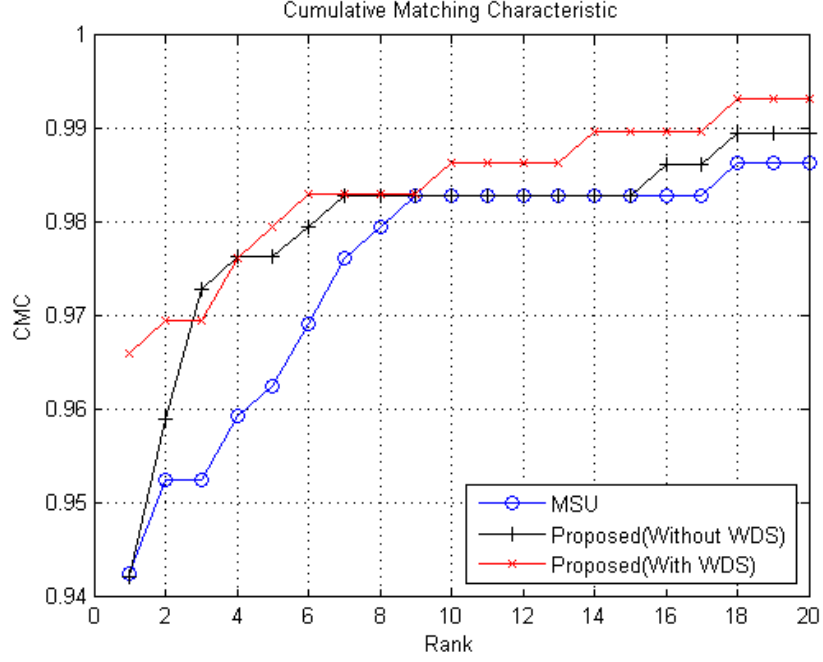


Fig. 3.24.: Retrieval performance of our proposed methods and MSU method using CMC in ROI dataset without background image

$$precision = \frac{\# \text{ of relevant gallery images on the retrieved images}}{\# \text{ of retrieved images}} \quad (3.32)$$

$$recall = \frac{\# \text{ of relevant gallery images on the retrieved images}}{\# \text{ of relevant images in gallery}} \quad (3.33)$$

Note that both metrics are functions of the number of retrieved images. To construct the precision and recall, 11-point Interpolated Average Precision [145] is used. Figure 3.25 and Figure 3.26 shows the precision and recall. This show that our method without WDS has higher precision in all recalls on both ROI datasets (with background and without background). Also, our method with WDS outperforms our method without WDS and the MSU method in terms of precision and recall.

Mean Average Precision (MAP) [145] is also used to evaluate our experiments. MAP is a single-value metric related to precision and recall. It is computed as the

Table 3.17.: CMC and MAP in ROI dataset with background images as reported in [11]

Method	Submission	CMC				MAP
		Rank 1	Rank 10	Rank 100	Rank 300	
CEA1	phase 1	0.731	0.737	0.771	0.811	0.733
CEA2	phase 2	0.781	0.785	0.818	0.845	0.783
Compass	phase 2	0.175	0.250	0.325	0.475	0.028
MITRE	phase 2	0.771	0.902	0.956	0.970	0.825
Morpho Track	phase 1	0.936	0.949	0.960	0.966	0.940
Morpho Track	phase 2	0.946	0.970	0.976	0.980	0.954
Purdue	phase 2	0.916	0.943	0.953	0.970	0.924

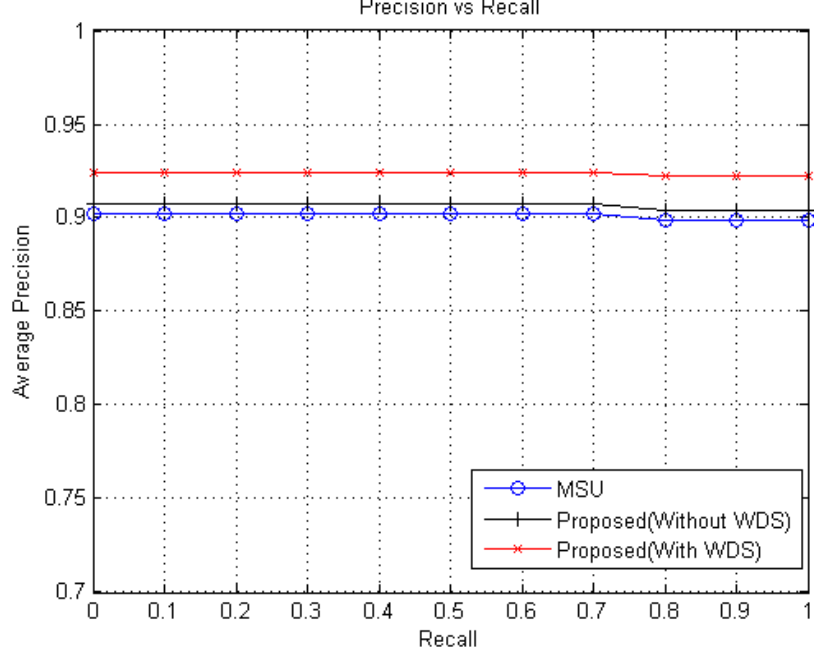


Fig. 3.25.: Retrieval performance of our proposed methods and MSU method using precision and recall for the ROI dataset with background images

mean of the average area under the precision and recall curve across all probe images. Therefore, a larger MAP means that the method is more accurate.

$$MAP(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} Precision(R_{jk}) \quad (3.34)$$

Where Q is a set of probe images, m_j is the recall level for the j^{th} probe image, and R_{jk} is the k^{th} recall level for the j^{th} probe image. In this experiment, we set $m_j=11$ for any j and $R_{jk} = 0, 0.1, 0.2, \dots, 1$. Table 3.16 and Table 3.15 show the MAP values for our methods and the MSU method. Our experimental results show that our method without WDS improves 0.36% and 0.53% more than the MSU method for both cases (with background images and without background images) respectively. Also, our method with WDS improves 2.82% and 2.25% more than the MSU method for both cases respectively.

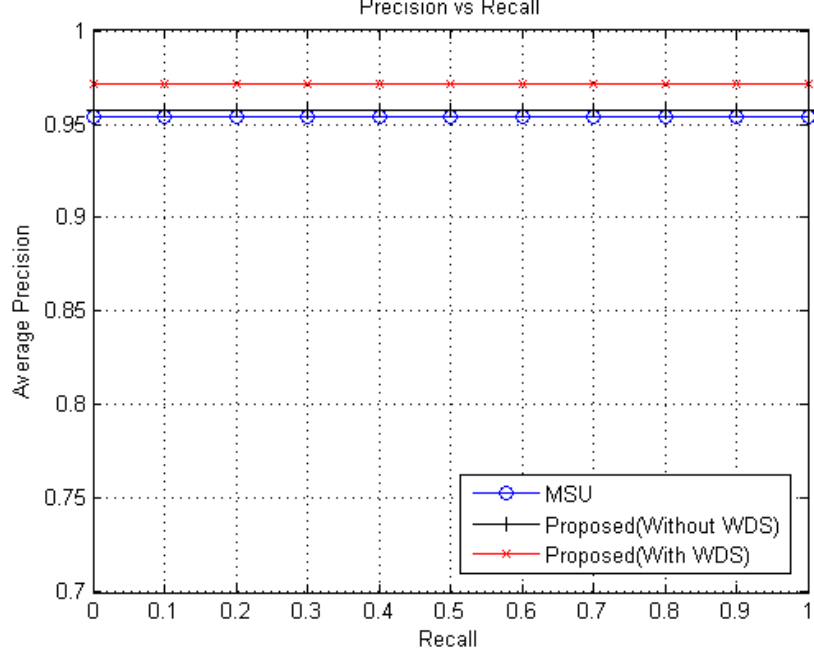


Fig. 3.26.: Retrieval performance of our proposed methods and MSU Method using precision and recall for the ROI dataset without background images

Our method (Purdue) also outperforms 4 other methods reported in the NIST challenge [11]. Note that our method reported in [11] is with WDS. NIST also evaluated all the submitted methods in terms of CMC, precision and recall, and MAP. Table 3.17 shows the comparison results. Note that submission in the table means which phase each method is submitted to. This table only considered the case that background images are added to the gallery images as “distractors”. Since we do not have the results of other methods reported in [11] we cannot show the precision and recall curves of the other methods in this thesis. This table shows that our method has higher CMC and MAP scores than the other 4 methods (CEA1, CEA2, Compass, and MITRE), but it has lower scores than Morpho Track’s method. Note that our method has a lower CMC score than MITRE by 0.3% in rank 100, but it has much higher CMC scores in rank 1 and rank 10 by 14.5% and 4.1% respectively. Therefore,

our method has higher MAP score (92.4%) than MITRE (82.5%) by 9.9%. More detailed CMC results are in [11]. Morpho Track’s submission to Tatt-C is higher than our method in terms of both CMC and MAP. Their phase 2 method achieved 94.6% and 97.0% CMC score in rank 1 and rank 10 respectively, and it outperformed our method in rank 1 and rank 10 by 3.0% and 2.7% respectively. They also achieved 95.4% MAP score and this is 3.0% higher than our method.

3.7 Conclusions and Future Work

We described new local and global descriptors for tattoo image retrieval. Our proposed descriptor is robust to partial shape distortions as well as invariant to translation, scale, and rotation. By combining robust similarity with our MHLC, we achieved more accurate results. A global shape descriptor combining MH and 2D Fourier Transform robust to rotation was proposed as well. Our experimental results showed our method achieved 70.3% top rank-1 accuracy and 88.36% top rank-10 accuracy in dataset 1, 59.84% top rank-1 accuracy and 88.36% top rank-10 accuracy in dataset 2, and 96.15% top rank-1 accuracy and 98.71% top rank-10 accuracy in dataset 4. It outperformed the method of [4] by 13.3%, 14.85% and 0.6% in top rank-1 and 1.98%, 8.32% and 1.90% in top rank-10 in dataset 1, 2 and 4 respectively. Also, our method outperformed not only the existing hand crafted methods but also the deep learning based method. Our experimental results showed that our RMHLC outperformed the deep learning based image retrieval method in [34] by 5.7% and 4.9% in top rank-10 and top rank-20.

Also we introduced the improved MHLC descriptor, called as DMHLC descriptor. Instead of using the spatial distribution of the SIFT features, the DMHLC descriptor uses the spatial distribution of the densely sampled features on the tattoo object to generate the multiple polar histograms. The multiple polar histograms are combined with the SIFT descriptor. Our experimental results showed that our DMHLC descriptor with the robust image similarity achieved 71.29% top rank-1 accuracy and 84.16%

top rank-10 accuracy in dataset 1, and 64.66% top rank-1 accuracy and 88.77% top rank-10 accuracy in dataset 2. It improved our MHLC descriptor with the robust image similarity by 3.96% and 6.45% in top rank-1 and 6.93% and 0.2% in top rank-10 in dataset 1 and 2 respectively.

We introduced our modified inductive matching to improve the image retrieval accuracy. By considering all the similarities between all the images in the database, the image retrieval accuracy was improved. Different from existing methods such as the diffusion process and the inductive matching, the modified inductive matching retrieve the most dissimilar M_2 database images to an input image first. Then, when the similarity between the input image and a database image is computed, the mean of the image similarities between the M_2 database images and the database image is subtracted from the pairwise similarity between the input image and the database image. Our experimental results showed the modified inductive matching with the combination of DMHLC descriptor and the robust image similarity achieved 67.78% top rank-1 accuracy and 89.81% top rank-10 accuracy in dataset 2. It improved the pairwise image similarity based on the combination of DMHLC descriptor and the robust image similarity by 3.12% in top rank-1 and 1.04% in top rank-10 dataset 2. We also showed that the modified inductive matching worked well with the other image descriptor or the other image similarity.

We introduce another image descriptor based on local self similarity (LSS) and SIFT. We also proposed the image similarity metric weighted by the distance of the descriptors for dataset 5. Our experimental results showed that our method achieved 91.56% top rank-1 accuracy and 94.25% top rank-10 accuracy. It outperformed the method of [4] by 2.7% in top rank-1 and 0.67% in top rank-10.

4. TATTOO IMAGE CLASSIFICATION BASED ON SPARSE CODING

In this chapter, we describe our spatial pyramid alignment technique for sparse coding based object classification [146]. Since our method can be used in general images as well as a tattoo image, we show our experimental results on the tattoo image dataset [61] as well as public image datasets such as Caltech 101 [147] and Caltech 256 [148]. Even though we mainly focus on the combination of our spatial pyramid alignment and sparse coding methods in this chapter, our method can be used in the deep learning based object classification that adopts a spatial pyramid pooling [149] in the deep neural network.

4.1 Review of Existing Methods

The bag of visual words (BOW) [46] model has been widely used for image representation. In the BOW model, each local feature is coded using a set of predefined codewords. The set of codewords is referred to as the codebook. Coding features using the codebook is analogous to representing vectors using a set of basis vectors. The codebook is usually constructed using a set of local image features randomly sampled from the training images. The feature coding vector is the output of the coding process. It is comprised of a set of coefficients where each coefficient is the contribution of a particular codeword in representing the feature. Vector quantization (VQ) [47, 48] simplifies the coding process by assuming that a feature can be represented by a single codeword. Therefore, all the elements of the coding vector are zeros except for a single element corresponding to the codeword closet to this feature. However, vector quantization is not adequate to represent the variation of features. This causes degradation in the performance of image representation and

classification. Sparse coding [49–59] has been utilized to address this problem. In sparse coding each local image feature is represented by a combination of a small number of codewords.

The final image representation is based on the coding vectors of all the local image features. To combine multiple coding vectors into a single vector, average or max pooling [60] is utilized. In average pooling, the final image representation vector is computed by averaging all the coding vectors, whereas max pooling uses the maximum value of each element among all the coding vectors separately. The basic BOW model combines the features coding vectors without considering the spatial distribution (layout) of the local image features. This means that the spatial locations of the local images features are not used in the BOW model. This drawback of the BOW model limits the descriptive power of the final image representation.

To address this problem, spatial pyramid feature pooling (SPP) [47, 49, 52–58] has been proposed and incorporated in most feature coding methods. To construct a spatial pyramid, an image is partitioned into $2^l \times 2^l$ subregions at different l^{th} levels ($l=0,1,2$). The first level consists of 16 subregions, whereas the second level contains 4 subregions and the third one is a single region. Instead of using max or average pooling on the entire image, SPP is done on each subregion. The final image representation of SPP is the concatenation of all the subregions representation vectors. Existing methods which are based on SPP assume that the center of an object is aligned with the center of the image. Therefore, the center of the image is used as the center of the spatial pyramid. However, the center of most images are not aligned with the center of objects correctly. This misalignment propagates in feature pooling results in several subregions at multiple pyramid levels.

There exist many image classification methods based on sparse coding. Codebook construction is essential to accurate image representation and classification. Many sparse coding methods have been proposed to learn accurate codebooks. In [58] a sparse representation for face recognition is shown to perform better than the conventional face recognition representations. In [59] the importance of an accurate

codebook generation is addressed. K-SVD is also introduced to learn a codebook accurately. In [51] discriminative K-SVD is proposed. K-SVD learns a codebook and a classifier jointly in an unsupervised approach, which results in an improved image classification accuracy. In [49] a supervised codebook learning method based on K-SVD is proposed. In [150] a kernel function is used to generate a codebook, and is shown to improve the classification accuracy. In [56], the hierarchical sparse coding is proposed. Feature coding and pooling are used on local image features sequentially to generate subregion representation vectors. The same feature coding method is used on the subregion representation vectors. This hierarchical coding achieves image representation robust to objects deformations. In [57] multiple image patches are used as local image features. These features are encoded using K-SVD based sparse coding in combination with a hierarchical sparse coding scheme.

In [47, 52, 151] the combination of the feature coding and the spatial pyramid is investigated. The classification performance is improved by combining the vector quantization with the spatial pyramid in [47]. In [52] the sparse coding is used with spatial pyramid instead of vector quantization. It shows that sparse coding with spatial pyramid can represent an image with more discriminative power. In [151] the importance of locality constrained feature coding (LLC) is addressed. The locality constraint is shown to be more powerful than the sparsity constraint in coding features.

In [53, 54] geometric feature pooling methods are described. Based on the assumption that the same class image shares similar spatial layouts, the weighted spatial pooling function is learned. In [152] the regions for feature pooling are learned. Instead of dividing the entire image using a spatial pyramid grid, a hierarchical structure of subregions is learned and is shown to improve the classification accuracy. In [55] a sparse coding method based on an ensemble of classifiers is proposed. This method is shown to be robust against overfitting.

Most existing methods described here use spatial pyramids and feature coding methods to improve the classification accuracy, However, the proper alignment between the center of an object and the center of the spatial pyramid is not addressed.

In this thesis, we propose a method to estimate the centers of various objects and propose to align the spatial pyramid center accordingly. Our proposed method is based on the spatial layout of the max pooled features. We also propose a final image representation descriptor robust to the misalignment of the spatial pyramid center. The experimental results demonstrate that our method is simple yet efficient in handling misalignments issues.

4.2 Center-Aligned Spatial Pyramid (CASP) Based Object Classification

Figure 4.1 shows the block diagram of our proposed object classification method. First, the dense local image features are extracted from an image. Coding vectors for all the local image features are generated using sparse coding. Instead of generating a regular spatial pyramid grid, we generate a Center-Aligned Spatial Pyramid (CASP) by estimating the center of the object in the image and aligning the center of the pyramid accordingly. Max pooling of all the feature coding vectors of the entire image is used to estimate the center of the object. Final image descriptor is found using max pooling in our CASP. Furthermore, to make our image descriptor robust to object deformations, we generate multiple CASP based image descriptors. Each descriptor is obtained by shifting the estimated center in a pre-defined margin. The maximum value of each element among all the image descriptors is computed to generate our final image descriptor. For image classification, the linear support vector machine (SVM) [153, 154] is used.

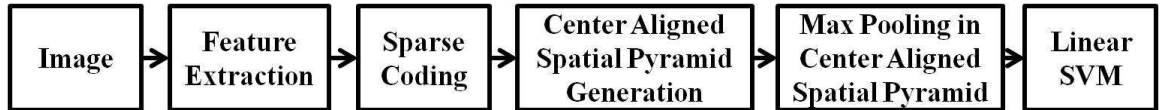


Fig. 4.1.: System overview

4.2.1 Sparse Coding and Max Pooling

To generate a feature coding vector for a local image feature, we use sparse coding in [52]. Once the local image feature is extracted from an image, the feature coding and codebook learning process is done by minimizing the following formula.

$$\begin{aligned} \underset{\mathbf{u}_m, \mathbf{V}}{\operatorname{argmin}} \quad & \sum_{m=1}^M \|\mathbf{f}_m - \mathbf{V}\mathbf{u}_m\|^2 + \lambda \|\mathbf{u}_m\| \\ \text{s.t.} \quad & \|\mathbf{v}_c\| \leq 1 \quad m = 1, 2, \dots, M \end{aligned} \quad (4.1)$$

where $\mathbf{f}_m \in \mathbb{R}^{d \times 1}$ is the m^{th} local image feature vector, $\mathbf{u}_m = [u_{1m}, u_{2m}, \dots, u_{cm}, \dots]^T \in \mathbb{R}^{C \times 1}$ is a feature coding vector for \mathbf{f}_m , $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_C] \in \mathbb{R}^{d \times C}$ is a codebook, \mathbf{v}_c is the c^{th} codeword in the codebook, M is the total number of the local image features in an image, and C is the size of the feature coding vector. For minimizing this, “feature-sign search” [155] is used. Once all the feature coding vectors for all the local image features are generated, max pooling is used to generate a subregion descriptor using all the feature coding vectors within the subregion in spatial pyramid. Let z_c^{ijl} be the max value of the c^{th} element of all the feature coding vectors within the $(i, j)^{th}$ subregion at the l^{th} pyramid level.

$$\begin{aligned} z_c^{ijl} &= \max(w_1^l |u_{c1}|, w_2^l |u_{c2}|, \dots, w_m^l |u_{cm}|, \dots, w_M^l |u_{cM}|) \\ w_m^l &= \begin{cases} 1 & \text{if } \mathbf{p}_m \in \mathbf{R}_{ij}^l \\ 0 & \text{otherwise} \end{cases} \quad i, j = 1, 2, \dots, 2^l \end{aligned} \quad (4.2)$$

where \mathbf{R}_{ij}^l is $(i, j)^{th}$ subregion at l^{th} level in pyramid, \mathbf{p}_m is the spatial coordinate of \mathbf{f}_m , and $\mathbf{z}^{ijl} = [z_1^{ijl}, z_2^{ijl}, \dots, z_C^{ijl}]$ is the subregion representation vector for \mathbf{R}_{ij}^l .

4.2.2 Center-Aligned Spatial Pyramid Generation

BOW models using dense local image features have demonstrated higher accuracy in object recognition compared to sparse features (i.e. interest points) [46, 156, 157]. One of the famous dense local image features is a Dense SIFT (Dense Scale Invariant Feature Transform). In Dense SIFT, single SIFT descriptor is generated on each

of densely sampled pixel locations. The single SIFT descriptor is used as a local image feature. Since the center of an object is difficult to estimate from dense local image features, the center of the spatial pyramid is assumed to be the center of the image. However, when considering multiple images containing the same object, the misalignments between the centers of the spatial pyramids and the object center cause parts of the object to lie in different subregions within the spatial pyramids. This misalignment propagates in feature pooling results in several subregions at multiple pyramid levels.

In Figure 4.2 two images for the same class (fish) illustrate this problem. The blue lines represent the grid of the original spatial pyramid and the red lines represent the grid of our CASP. As shown in Figure 4.2, our CASP splits the same parts of two object into the same subregions, but the original spatial pyramid does not. To find

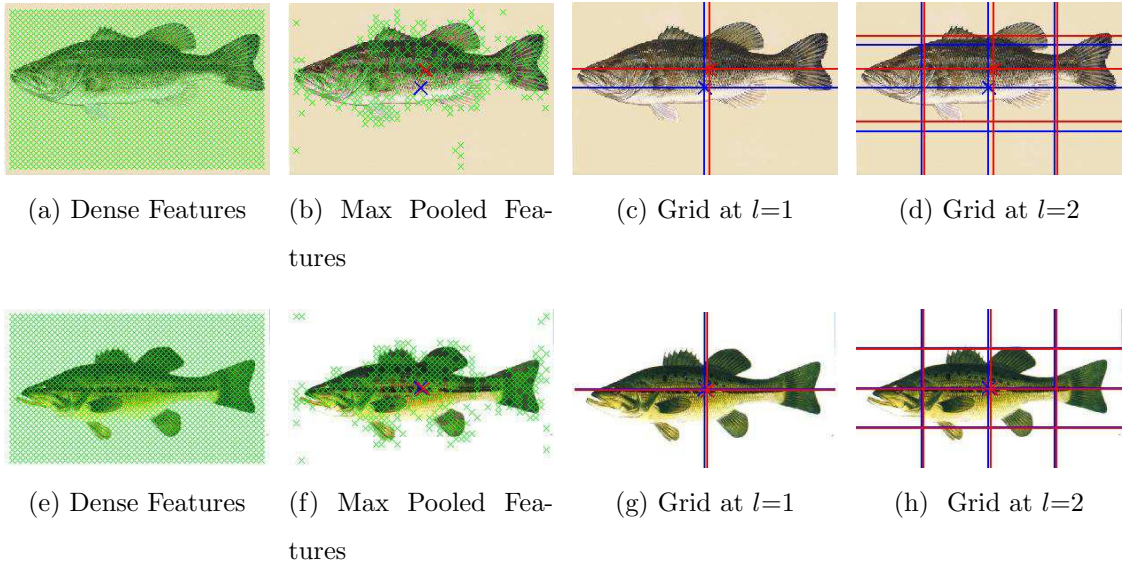


Fig. 4.2.: Example of Center-Aligned Spatial Pyramid Generation: Blue color represents the original spatial pyramid grid and red color represents our CASP grid

the center of an object, we describe a simple but efficient method based on max pooled features on the entire image. Since the max pooled features indicates the most salient parts of an object, the spatial layout of the max pooled features can

be used to estimate the center of an object. We find the index of the most salient feature, m' that has the maximum response to the c^{th} codeword in \mathbf{V} :

$$m' = \underset{m}{\operatorname{argmax}} (w_m^0 |u_{cm}|) \quad m = 1, 2, \dots, M \quad (4.3)$$

where $m' = 1, 2, \dots, M'$, M' is the number of the max pooled features in the entire image. Note that we repeat this process from $c=1, \dots, C$, and we do not find the max pooled feature index, m' if the maximum response for c^{th} codeword is zero. Therefore, the number of max pooled features, M' cannot be larger than the feature coding vector size, C . Then, the center of an object, \mathbf{p}_{ct} is estimated as follows:

$$\mathbf{p}_{ct} = \sum_{m'=1}^{M'} \mathbf{p}_{m'} \quad \text{where } \mathbf{p}_{ct} = (x_{ct}, y_{ct}) \quad (4.4)$$

Once the center of a spatial pyramid is estimated, our CASP is constructed based on subregions of it defined as:

$$\begin{aligned} \mathbf{R}_{ij}^l &= \{(x, y) | x_l < x \leq x_h \text{ and } y_l < y \leq y_h\} \\ x_l &= \frac{x_{ct}}{2^{l-1}}(i-1) & i \leq 2^{l-1} \\ x_h &= \frac{x_{ct}}{2^{l-1}}i \\ x_l &= \frac{w-x_{ct}}{2^{l-1}}i + 2x_{ct} - w & i > 2^{l-1} \\ x_h &= \frac{w-x_{ct}}{2^{l-1}}(i+1) + 2x_{ct} - w \\ y_l &= \frac{y_{ct}}{2^{l-1}}(j-1) & j \leq 2^{l-1} \\ y_h &= \frac{y_{ct}}{2^{l-1}}j \\ y_l &= \frac{h-y_{ct}}{2^{l-1}}j + 2y_{ct} - h & j > 2^{l-1} \\ y_h &= \frac{h-y_{ct}}{2^{l-1}}(j+1) + 2y_{ct} - h \end{aligned} \quad (4.5)$$

where h and w are the image height and width, respectively. The final image descriptor is the concatenation of all the max pooled feature vector over all the subregions in the CASP centered at \mathbf{p}_{ct}

$$D_{\mathbf{p}_{ct}} = [\mathbf{z}^{ijl}] \quad i, j = 1, 2, \dots, 2^l, l = 0, 1, 2 \quad (4.6)$$

Note that the c^{th} element of $D_{\mathbf{p}_{ct}}$ is obtained by finding the maximum value among all the c^{th} elements of all the feature coding vectors within the $(i, j)^{th}$ subregion at

the l^{th} pyramid level. We also propose to add new subregions. These subregions are formed by connecting the centers of the four nearest subregions at each level of CASP. Figure 4.3 illustrates this concept. In Figure 4.3 left one is our previous CASP and right one is our modified CASP (MCASP). Note that the newly added subregions (shown in red) describe the parts of object more in detail.

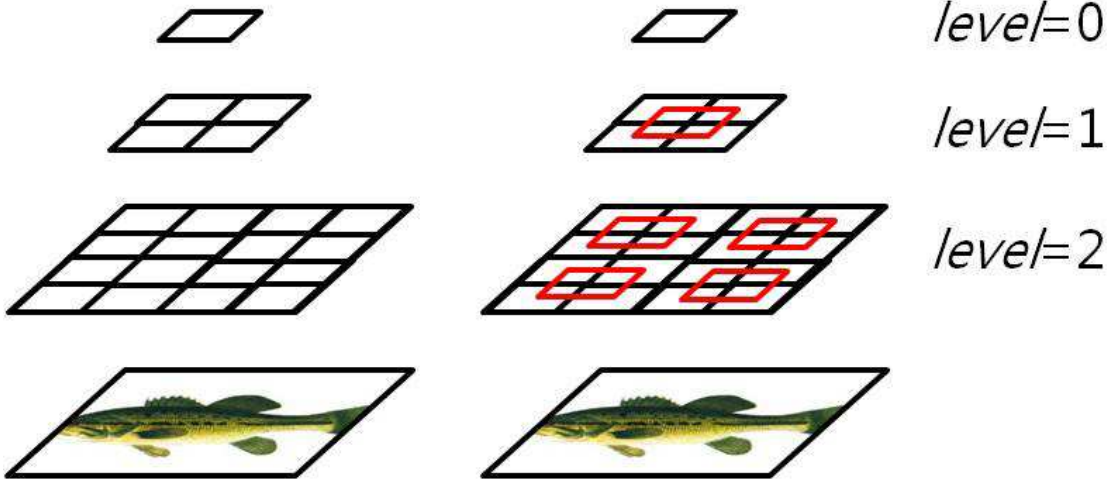


Fig. 4.3.: Our Modified Center-Aligned Spatial Pyramid (MCASP) structure

Let the five subregion descriptors additionally added at $l = 1, 2$ be \mathbf{z}_{ad1}^1 , \mathbf{z}_{ad1}^2 , \mathbf{z}_{ad2}^2 , \mathbf{z}_{ad3}^2 , and \mathbf{z}_{ad4}^2 and the concatenation of these descriptors be $D_{\mathbf{pct}}^{add}$. Then, our image descriptor is generated by concatenating $D_{\mathbf{pct}}$ and $D_{\mathbf{pct}}^{add}$.

$$D'_{\mathbf{pct}} = [D_{\mathbf{pct}}, D_{\mathbf{pct}}^{add}] = [d_1^{\mathbf{pct}}, d_2^{\mathbf{pct}}, \dots] \quad (4.7)$$

4.2.3 Image Descriptor Robust to Object Deformation

Even though our proposed CASP is aligned with the center of the object of interest, object deformations might still occur and cause degradation in performance. When an object is deformed, its parts might be shifted from one subregion to an adjacent one. This is referred to spatial quantization error in spatial pyramid. To solve this spatial quantization error, we shift \mathbf{p}_{ct} multiple times, and for each time we

find the image descriptor, $D'_{\mathbf{p}_{\text{ct}}}$ centered on the shifted \mathbf{p}_{ct} . The range of shifting is referred to the margin of deformation. Then, the final image descriptor is generated based on max pooling of each element of the multiple image descriptors, $D'_{\mathbf{p}_{\text{ct}}}$.

$$FD_{\mathbf{p}_{\text{ct}}} = [\max(d_1^{\mathbf{p}_{\text{ct}}+\mathbf{p}_{\text{margin}}}), \max(d_2^{\mathbf{p}_{\text{ct}}+\mathbf{p}_{\text{margin}}}), \dots] \quad (4.8)$$

where $\mathbf{p}_{\text{margin}} = (x_{\text{margin}}, y_{\text{margin}})$, $-t \leq x_{\text{margin}} \leq t$, and $-t \leq y_{\text{margin}} \leq t$. This enables us to capture the similar salient parts of an object in the same subregions of the same class images even when there are small shape variations between them.

4.3 Experimental Results

To evaluate the performance of our method, we implemented our method on two different feature coding methods: SC [52] and LLC [125]. In this experiment three different datasets are used: Caltech-101 [147], Caltech-256 [148], and Evil Tattoo dataset [158]. First two datasets are popular for testing object recognition, and the third dataset is suitable to evaluate the robustness of an image descriptor against object deformations.

We compared our methods against several existing methods that use the BOW model combined with a spatial pyramid. Additionally, for SC [52] and LLC [125] we reported their results based on a local evaluation using the software they provide under the exact same experimental setup: the same sets of images randomly chosen for training and test, and the same codebook. These experimental results are referred to SC (our evaluation) and LLC (our evaluation). Note that the accuracies of our evaluation for SC and LC is slightly different from what they reported in their publications [52, 125]. For fair comparison, we compare our proposed methods against SC (our evaluation) and LLC (our evaluation) since the same experimental setup is used.

We refer to the method based on SC for feature coding and CASP for spatial pyramid as SC+CASP. SC+MCASP is the method that uses SC with our modified CASP as shown in Figure 4.3. LLC+CASP is the method that uses LLC as feature coding and CASP as spatial pyramid. LLC+MCASP is the method that uses LLC

with our modified CASP as shown in Figure 4.3. In both methods, the final image descriptor is generated by shifting the estimated center within a pre-defined margin, $t = 3$. For local image features, dense SIFT in [52, 125] are used for all the datasets. For Caltech-101 and Caltech-256, we followed the same experimental setting as in [52, 125]. We randomly select 30 images per class for training and use the remaining images for testing. After computing classification accuracy for each object class, the mean classification accuracy over all classes is computed. We repeat this process 5 times and compute the average accuracy.

4.3.1 Caltech-101 Dataset

The Caltech-101 dataset [147] contains 9144 images from 102 different class, including 101 object class and 1 additional background class. The number of image per class ranges from 31 to 800. For this dataset, we followed the same experiment setting as [52]. The size of a codebook for SC and LLC is set to $C = 1024$. Table 4.1 shows the experimental results. With SC, our method (SC+MCASP) outperforms other existing methods except MHMP. Our method (SC+MCASP) outperforms the original method, SC by 2.74%. Also, our method (LLC+MCASP) outperforms the original method, 2.08%. These results show that our MCASP can be combined with two different feature coding methods and it improves both of original methods.

4.3.2 Caltech-256 Dataset

The Caltech-256 dataset [148] contains 29,780 images from 257 different class, including 256 object class and 1 additional background class. The number of image per class ranges from 80 to 827. The size of a codebook for SC and LLC is set to $C = 1024$ and $C = 4096$ as in [52, 125]. Table 4.1 also shows the experimental results for Caltech-256. With SC and LLC, our method (SC+MCASP) and our method (LLC+MCASP) outperforms other existing methods except MHMP as well.

Table 4.1.: Classification accuracy (%) on Caltech-101 and Caltech-256 datasets

Method	Caltech-101	Caltech-256
KC [150]	64.16	27.17
VQ [47]	64.60	29.51
SRC [58]	70.70	33.33
D-SVD [159]	73.00	32.67
SC [52]	73.20	34.02
LLC [151]	73.44	41.19
LC-KSVD [49]	73.60	34.42
HSC [56]	74.00	N/A
MHMP [57]	82.50	50.70
LLC (our evaluation)	71.95	35.96
Our method (LLC+CASP)	73.21	36.53
Our method (LLC+MCASP)	74.02	37.55
SC (our evaluation)	72.33	34.75
Our method (SC+CASP)	74.50	36.30
Our method (SC+MCASP)	75.07	37.09

Table 4.2.: Classification accuracy (%) on Evil Tattoo dataset

Method	Accuracy (%)
LLC (our evaluation)	51.57
Our method (LLC+CASP)	51.93
Our method (LLC+MCASP)	52.33
SC (our evaluation)	54.12
Our method (SC+CASP)	55.81
Our method (SC+MCASP)	56.56

Our method (SC+MCASP) outperforms the original SC by 2.34%, and our method (LLC+MCASP) outperforms the original LLC by 1.59%.

4.3.3 Evil Tattoo Dataset

The evil tattoo dataset [158] contains 1,477 images in total from 27 different class. All the images in this dataset are acquired from eviltattoo.com. The number of image per class ranges from 14 to 180. For this dataset, we randomly select 10 images per class for training and use the rest of images for testing. The size of a codebook for SC and LLC is set to $C = 1024$. Table 4.2 also shows the experimental results for the evil tattoo dataset. Since there are no published results on this dataset, we compared our method against our local evaluations of SC and LLC. With SC, our method (SC+MCASP) outperforms the original SC by 2.44%. With LLC, our method (LLC+MCASP) outperforms the original LLC by 0.76%.

4.3.4 Discussion

We tested our method with two different feature coding methods on three different dataset. Our experimental results show that by combining our proposed spatial

pyramid (MCASP) with the feature coding methods the classification accuracies are improved. This was demonstrated using the three challenging datasets. We also investigated the impact of using the proposed new subregions within MCASP. This was achieved by comparing the improvement achieved using CASP versus MCASP. The experimental results demonstrated that the improvement is equally attributed to CASP and MCASP. For example, LLC+MCASP improved LLC+CASP by 1% on the Caltech-256 dataset, where as LLC+CASP improved LLC by 0.6%. Also, SC+MCASP improved SC+CASP by 0.8% on the Caltech-256 dataset, where as SC+CASP improved SC by 1.55%. This means that both contributions: spatial pyramid alignment and the introduction of new subregions, are significant to the improvement in classification accuracy.

4.4 Conclusions and Future Work

In this thesis we proposed a simple but efficient spatial pyramid alignment method that can be combined with the existing feature coding methods. By using max pooled features, we estimate an object center and align the spatial pyramid accordingly. We also propose an image representation descriptor robust to misalignment and object deformations using max pooling on multiple image descriptors generated by shifting the pyramid center in a pre-defined margin. We tested the modified center-aligned spatial pyramid with two different feature coding methods on three different datasets. Our experimental results show that by combining our proposed spatial pyramid (MCASP) with the feature coding, classification accuracy is improved. In the future, we will investigate methods to estimate an object center when there is background clutter.

5. SHAPE MATCHING AND RETRIEVAL USING A SELF SIMILAR AFFINE INVARIANT DESCRIPTOR

In this chapter, we describe our shape matching and retrieval technique using a self similar affine invariant descriptor [139].

Shape matching has been widely used in computer vision and is critical in image retrieval. A better representation of the shape of an object can greatly improve matching accuracy. It is difficult to have an accurate shape model because geometric transformation such as rotation, translation, scaling, shearing, and deformations such as noise, articulation, and occlusion are not easy to be modeled.

There are mainly two types of shape representations, contour-based and region-based models [160, 161]. Region-based representation uses all the points on the contour of the shape of an object as well as the points inside the contour [161]. Contour based representation uses information from the shape boundary and achieves good performance [162, 163]. Two general types of shape descriptors are used with a contour-based representation: global shape descriptors and local shape descriptors. Since non-rigid deformations are difficult to be modeled using a global shape descriptor, many existing methods have focused on local shape descriptors with consideration of global information [164–168]. Therefore we focus on a local shape descriptor.

Since non-rigid deformation can be approximated by a local affine transformation [125], we describe in this thesis a local shape descriptor invariant to affine transform. Our contribution is a local affine invariant shape descriptor based on self similarity of the object shape (SSAI). The SSAI is also insensitive to local shape distortion and articulation. By using multiple SSAI descriptors based on different sets of neighbor points, object shapes deformed by non-rigid deformation can be recognized more accurately. We also describe an efficient image matching approach using multiple SSAI descriptors.

5.1 Review of Existing Shape Methods

Many methods have been proposed to combine both characteristics of global and local shape information. In [117], a Shape Context (SC) descriptor is described to exploit spatial correspondence between two shapes. The spatial correspondence was used to construct an aligning transformation in order to map one shape to the other. The log polar histogram was used to generate the SC descriptor. The dissimilarity between shapes can be then estimated as the sum of matching errors using alignment errors. In [169] the affine transformation and a non-rigid local transformation were combined to generate a probability model that determines similarity between shapes. Softassign [170] was used for this model since it is equivalent to minimizing the EM free energy function. Unfortunately, this method only allowed specific representation of shapes. In [171] SC was enhanced by using an ordered Bag of Feature model [172]. In [126] a representation based on strings of symbols for shape matching was used. In both [171] and [126] dynamic programming was used to improve the matching performance. For the SC descriptor to adapt to articulation, such as non-linear transformation, the inner distance was introduced to replace Euclidean distance in [164]. The shape boundary is assumed to be known. The inner distance does handle occlusion and large deformation well due to its sensitivity to shape topology. In [160] pairwise geometric relations between contour fragments was used to establish self-closed partial descriptor that is invariant to rotation, scaling, and translation. This method is able to capture local information while achieving good performance against some deformations such as occlusion and distortion. In both [165] and [166] a closed contour shape was divided into curve segments hierarchically. These segments were compared explicitly for shape matching. In [167] contour flexibility was proposed to represent the deformable potentials at contour point. This descriptor has more resistance to deformation. In [125] a local affine invariant descriptor was described. An orthogonal projection matrix for a set of contour points is used to construct the affine-invariant descriptor. In [161] a representation based on height functions

of contour points was introduced. Every contour point was described by a height function that is defined based on the distances from other points to this point's tangent line. The height descriptor is acquired by smoothing the height functions since a precise description may be too sensitive to local deformations. In [163] a metric partition constraint was described and motivated by the idea of height functions to bridge the local and global information while retaining good computational efficiency. Metric sequence with Euclidean distance and triangle radius were used as two different distance metrics. The author used a partition and smoothing process to reduce the description's dimension as well as sensitivity to deformations.

5.2 Proposed Shape Retrieval System

5.2.1 System Overview

Figure 5.1 shows the block diagram of our proposed shape retrieval system. In the archive process for each image in our database we obtain contour points and sort them in clockwise order. From these points we obtain sets of Self Similarity Based Affine Invariant (SSAI) descriptors. In the retrieval process when an input image is presented to our system, we first generate SSAI descriptors on contour points sorted in clockwise order similar to the archive process. We then compare the SSAI descriptors obtained from the input image against those obtained from images in our database using image matching method. The matching method first returns a score to retrieve the top M matched images. Then, the scores between top M matched images and database images are computed. Last, top N_{sim} images are retrieved based on the summation of those scores.

5.2.2 Self Similar Affine Invariant (SSAI) Descriptor

Since nonrigid deformation, perspective projection, and articulated motion can be approximated by a locally affine transformation [125], our descriptor can also

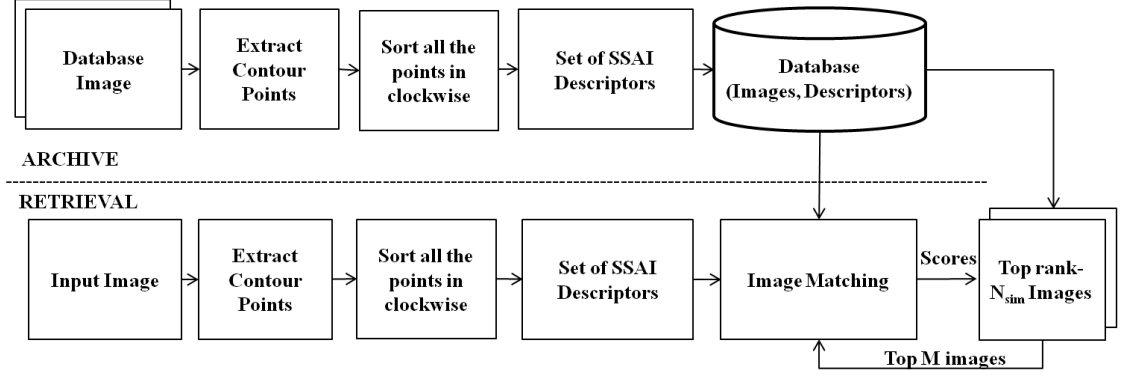


Fig. 5.1.: Our proposed shape retrieval system

represent an object shape properly under these deformations. If there are two sets of points transformed by an affine transform, subsets of each set of points are also related by the same affine transformation. Using this property with the pseudo inverse, we can construct a shape descriptor that is invariant to affine transformation. To generate a set of SSAI descriptors, we obtain contour feature points of an object from an image and sort them in clockwise order. In this thesis, the sorting method in [164] was used. Let X and Y be the sorted feature points on the contour of an input image (I_X) and a database image (I_Y):

$$X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_i, \dots, \mathbf{x}_N] \quad Y = [\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_j, \dots, \mathbf{y}_N] \quad (5.1)$$

where $\mathbf{x}_i = [x_{i1} \ x_{i2}]^T$ and $\mathbf{y}_j = [y_{j1} \ y_{j2}]^T$ are the points on the contour.

For \mathbf{x}_i and \mathbf{y}_j we obtain their neighbor points, X_i^k and Y_j^k ($k = 1, 2$).

$$\begin{aligned} X_i^k &= [\mathbf{x}_{i-n_k/2}, \mathbf{x}_{i+1-n_k/2}, \dots, \mathbf{x}_i, \dots, \mathbf{x}_{i-1+n_k/2}, \mathbf{x}_{i+n_k/2}] \\ Y_j^k &= [\mathbf{y}_{j-n_k/2}, \mathbf{y}_{j+1-n_k/2}, \dots, \mathbf{y}_j, \dots, \mathbf{y}_{j-1+n_k/2}, \mathbf{y}_{j+n_k/2}] \end{aligned} \quad (5.2)$$

where $n'_k = n_k + 1$ is the number of points in X_i^k and Y_j^k . Note that X_i^2 and Y_i^2 are subsets of X_i^1 and Y_i^1 respectively ($n_2 < n_1$). If all the points of X_i^1 are mapped onto

all the points of Y_j^1 by the affine matrix, H , then all the points of X_i^2 are also mapped onto all the points of Y_j^2 by H . Y_j^k can then be represented as

$$\begin{aligned} \mathbf{Y}_j^k &= H_k \mathbf{X}_i^k P_k = H \mathbf{X}_i^k P_k \\ \text{s.t. } \mathbf{Y}_j^k &= \begin{bmatrix} Y_j^k \\ \mathbf{1}_{n'_k}^T \end{bmatrix} \quad \mathbf{X}_i^k = \begin{bmatrix} X_i^k \\ \mathbf{1}_{n'_k}^T \end{bmatrix} \quad H = \begin{bmatrix} A & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \end{aligned} \quad (5.3)$$

where P_k is a $n'_k \times n'_k$ permutation matrix, $\mathbf{1}_{n'_k} = [1, \dots, 1]^T$, $\mathbf{t} = [t_1 \ t_2]^T$, $\mathbf{0} = [0 \ 0]^T$, and A is a 2×2 non-singular matrix. As shown in [125], (5.3) can be rewritten using $\tilde{Y}_j^k = Y_j^k C$, $\tilde{X}_i^k = X_i^k C$ and A instead of \mathbf{Y}_j^k , \mathbf{X}_i^k and H by multiplying \mathbf{Y}_j^k and \mathbf{X}_i^k to the centering matrix, C .

$$\begin{aligned} \tilde{Y}_j^k &= A \tilde{X}_i^k P_k \\ \text{s.t. } C &= (I - \frac{1}{n'_k} \mathbf{1}_{n'_k} \mathbf{1}_{n'_k}^T) \end{aligned} \quad (5.4)$$

where I is identity matrix. Note that \tilde{Y}_j^k and \tilde{X}_i^k are invariant under translation. Since \tilde{Y}_j^1 and \tilde{Y}_j^2 are transformed from \tilde{X}_i^1 and \tilde{X}_i^2 by the same A respectively, the relationship between \tilde{Y}_j^1 and \tilde{Y}_j^2 is consistent with the relationship between \tilde{X}_i^1 and \tilde{X}_i^2 under any matrix A . Under the assumption that \tilde{X}_i^k is a full row rank matrix pseudo inverse [173] of \tilde{Y}_j^1 multiplied by \tilde{Y}_j^2 is invariant to A .

$$\tilde{Y}_j^{1+} \tilde{Y}_j^2 = (A \tilde{X}_i^1 P_1)^+ (A \tilde{X}_i^2 P_2) \quad (5.5)$$

where $+$ is a pseudo inverse operator. Since any permutation matrix has orthonormal rows, A is non-singular, and \tilde{X}_i^k is a full row rank matrix, (5.5) can then be written as

$$\begin{aligned} \tilde{Y}_j^{1+} \tilde{Y}_j^2 &= P_1^+ \tilde{X}_i^{1+} A^+ (A \tilde{X}_i^2 P_2) = P_1^{-1} \tilde{X}_i^{1+} A^{-1} (A \tilde{X}_i^2 P_2) \\ &= P_1^T \tilde{X}_i^{1+} \tilde{X}_i^2 P_2 \end{aligned} \quad (5.6)$$

For permutation invariance, $\tilde{Y}_j^{1+} \tilde{Y}_j^2$ is first multiplied by its transpose.

$$\begin{aligned} R_{\tilde{Y}_j^{1+} \tilde{Y}_j^2} &= \tilde{Y}_j^{1+} \tilde{Y}_j^2 (\tilde{Y}_j^{1+} \tilde{Y}_j^2)^T = P_1^T \tilde{X}_i^{1+} \tilde{X}_i^2 P_2 (P_1^T \tilde{X}_i^{1+} \tilde{X}_i^2 P_2)^T \\ &= P_1^T \tilde{X}_i^{1+} \tilde{X}_i^2 (\tilde{X}_i^{1+} \tilde{X}_i^2)^T P_1 = P_1^T R_{\tilde{X}_i^{1+} \tilde{X}_i^2} P_1 \end{aligned} \quad (5.7)$$

Since $Y_j^{1+} \tilde{Y}_j^2$ can be infinite, it is computed using regularization as

$$Y_j^{1+} \tilde{Y}_j^2 = Y_j^{1+T} (\tilde{Y}_j^1 \tilde{Y}_j^{1T} + \lambda I)^{-1} \tilde{Y}_j^2 \quad (5.8)$$

where $\lambda = 1$ is a regularization parameter. Then, the diagonal elements of $Y_j^{1+} \tilde{Y}_j^2 (Y_j^{1+} \tilde{Y}_j^2)^T$ are obtained as

$$\text{diag}(R_{Y_j^{1+} \tilde{Y}_j^2}) = \text{diag}(P_1^T R_{X_i^{1+} \tilde{X}_i^2} P_1) = P_1^T \text{diag}(R_{X_i^{1+} \tilde{X}_i^2}) \quad (5.9)$$

where $\text{diag}(\cdot)$ is an operator to compute diagonal elements of input matrix. Since all the points in Y_j^{1+} and X_i^{1+} are already ordered in clockwise, P_1 is an identity or reverse identity matrix. Note that if an input image and a database image are mirrored, P_1 will be a reverse identity matrix. Otherwise, P_1 is an identity matrix. Then, our first descriptor for \mathbf{x}_i invariant to permutation and affine transformation is generated by convolving $\text{diag}(R_{X_i^{1+} \tilde{X}_i^2})$ and $\text{diag}(R_{X_i^{1+} \tilde{X}_i^2})$ multiplied by a reverse identity matrix, E .

$$Des_1^{\mathbf{x}_i} = \text{diag}(R_{X_i^{1+} \tilde{X}_i^2}) * E \text{diag}(R_{X_i^{1+} \tilde{X}_i^2}) \quad (5.10)$$

where $*$ is convolution operator and $Des_1^{\mathbf{x}_i}$ is the first descriptor for \mathbf{x}_i . Similarly our second descriptor for \mathbf{x}_i , $Des_2^{\mathbf{x}_i}$ is obtained as

$$Des_2^{\mathbf{x}_i} = \text{diag}(R_{X_i^{2+} \tilde{X}_i^1}) * E \text{diag}(R_{X_i^{2+} \tilde{X}_i^1}) \quad (5.11)$$

Note that $Des_1^{\mathbf{x}_i} = [d_1^1, d_1^2, \dots, d_1^{2n'_1-1}]$ is a $2n'_1-1$ dimensional vector and $Des_2^{\mathbf{x}_i} = [d_2^1, d_2^2, \dots, d_2^{2n'_2-1}]$ is a $2n'_2-1$ dimensional vector. Even though $Des_1^{\mathbf{x}_i}$ and $Des_2^{\mathbf{x}_i}$ are invariant to affine transformation, they can still be sensitive to local shape distortion. A smoothing process is used to make the descriptors insensitive to local shape distortion. Let the smoothed descriptors be $Des_{s1}^{\mathbf{x}_i} = [sd_1^1, sd_1^2, \dots, sd_1^{n'_1}]$ and $Des_{s2}^{\mathbf{x}_i} = [sd_2^1, sd_2^2, \dots, sd_2^{n'_2}]$. Then, $sd_1^{t'}$ and $sd_2^{t'}$ are obtained as

$$\begin{aligned} sd_1^{t'} &= \begin{cases} \frac{1}{2} \sum_{t=2t'-1}^{2t'} d_1^t & \text{if } t' \neq n'_1 \\ d_1^t & \text{if } t' = n'_1 \end{cases} \\ sd_2^{t'} &= \begin{cases} \frac{1}{2} \sum_{t=2t'-1}^{2t'} d_2^t & \text{if } t' \neq n'_2 \\ d_2^t & \text{if } t' = n'_2 \end{cases} \end{aligned} \quad (5.12)$$

Note that the lengths of the smoothed descriptors ($Des_{s1}^{x_i}$ and $Des_{s2}^{x_i}$) are also reduced from $2n'_1-1$ and $2n'_2-1$ to n'_1 and n'_2 respectively. Our SSAI descriptor is then defined by concatenating two smoothed descriptors with weight, w_d .

$$FDES^{x_i} = \begin{bmatrix} Des_{s1}^{x_i} \\ w_d Des_{s2}^{x_i} \end{bmatrix} \quad (5.13)$$

where $w_d = ||Des_{s1}^{x_i}||_2 / ||Des_{s2}^{x_i}||_2$. Our final SSAI descriptor, $FNDES^{x_i}$ is obtained by normalization

$$FNDES^{x_i} = FDES^{x_i} / ||FDES^{x_i}||_2 \quad (5.14)$$

5.2.3 Multiple SSAI Descriptors/Multiple Levels

The SSAI descriptor for one contour point is a local shape descriptor based on its neighbor points. To recognize an object more distinguishably, local shape descriptors of different sizes representing different local object shapes are used. In the SSAI descriptor the size of neighbor points is a parameter used to represent different local object shapes. Therefore, we change the size of neighbor points, n_1 and n_2 from small to large and generate multiple SSAI descriptors for one contour point, x_i .

$$n_{1\ell} = N_{lowest}\ell, \quad n_{2\ell} = 0.5N_{lowest}\ell \quad for \quad \ell = 1, 2, \dots, L \quad (5.15)$$

where $n_{1\ell}$ is the number of neighbor points (n_1) at ℓ^{th} level, N_{lowest} is the number of neighbor points at the lowest level, and $L \leq N/N_{lowest}$ is the number of levels. Since large local shape represented by large number of neighbor points can have more shape variations than small local shape, the first and second descriptor in (5.10) and (5.11) are re-defined at ℓ^{th} level with additional smoothing process.

$$\begin{aligned} Des_1^{x_i} &= diag_s(R_{X_i^1 + \tilde{X}_i^2}) * Ediag_s(R_{X_i^1 + \tilde{X}_i^2}) \\ Des_2^{x_i} &= diag_s(R_{X_i^2 + \tilde{X}_i^1}) * Ediag_s(R_{X_i^2 + \tilde{X}_i^1}) \end{aligned} \quad (5.16)$$

where $diag_s(\cdot)$ is a smoothed version of $diag(\cdot)$

$$\begin{aligned} sa_1^{t'} &= \begin{cases} \frac{1}{\ell} \sum_{t=\ell(t'-1)+1}^{\ell t'} a_1^t & \text{if } t' \neq N_{lowest} + 1 \\ a_1^t & \text{if } t' = N_{lowest} + 1 \end{cases} \\ sa_2^{t'} &= \begin{cases} \frac{1}{\ell} \sum_{t=\ell(t'-1)+1}^{\ell t'} a_2^t & \text{if } t' \neq 0.5N_{lowest} + 1 \\ a_2^t & \text{if } t' = 0.5N_{lowest} + 1 \end{cases} \end{aligned} \quad (5.17)$$

where a_1^t and a_2^t the t^{th} are elements of $diag(R_{\tilde{X}_i^1 + \tilde{X}_i^2})$ and $diag(R_{\tilde{X}_i^2 + \tilde{X}_i^1})$. Also, $sa_1^{t'}$ and $sa_2^{t'}$ are the t'^{th} elements of $diag_s(R_{\tilde{X}_i^1 + \tilde{X}_i^2})$ and $diag_s(R_{\tilde{X}_i^2 + \tilde{X}_i^1})$. For example, if $N_{lowest} = 20$, the length of $diag(R_{\tilde{X}_i^1 + \tilde{X}_i^2})$ is 21, 41, 61 at $\ell = 1, 2, 3$, and the length of $diag(R_{\tilde{X}_i^2 + \tilde{X}_i^1})$ is 11, 21, 31 at $\ell = 1, 2, 3$. After smoothing, the lengths of $diag_s(R_{\tilde{X}_i^1 + \tilde{X}_i^2})$ and $diag_s(R_{\tilde{X}_i^2 + \tilde{X}_i^1})$ at any ℓ are 21 and 11 respectively.

5.2.4 Image Matching

Given two points, \mathbf{x}_i and \mathbf{y}_j that belong to contour feature points in X and Y , the cost function between two points at ℓ^{th} level is defined as:

$$c_\ell(\mathbf{x}_i, \mathbf{y}_j) = \|FNDES_\ell^{\mathbf{x}_i} - FNDES_\ell^{\mathbf{y}_j}\|_2^2 \quad (5.18)$$

where $i, j = 1, 2, \dots, N$. If $c_\ell(\mathbf{x}_i, \mathbf{y}_j)$ is greater than threshold $\tau_\ell = \tau/\ell$, $c_\ell(\mathbf{x}_i, \mathbf{y}_j)$ is set to τ_ℓ . $\tau=0.15$ is determined empirically. For matching between X and Y we need to find the pairs of \mathbf{x}_i and \mathbf{y}_j that minimize the following cost function, $C_\ell(\pi_\ell)$ at ℓ^{th} level:

$$C_\ell(\pi_\ell) = \sum_{i=1}^N c_\ell(\mathbf{x}_i, \mathbf{y}_{\pi_\ell(i)}) \quad (5.19)$$

where $(i, \pi_\ell(i))$ is the pair of (i, j) minimizing $C_\ell(\pi_\ell)$. Since the points in X and Y are already ordered, the dynamic programming approach in [164] was used to find the pair of points, $(i, \pi_\ell(i))$ at each ℓ^{th} level. By summing minimum $C_\ell(\pi_\ell)$ from $\ell = 1$ to $\ell = L$, the matching cost between X and Y using multiple SSAI descriptors is obtained. Similar to [161], shape complexity is additionally used for the matching cost to improve retrieval accuracy. The idea of shape complexity is that simple shapes

are not easy to be distinguished as complex shapes. By using this idea, we improve the matching accuracy by assigning more weights to the simple shapes.

$$C_M(X, Y) = \sum_{l=1}^L \frac{C_l(\pi_l)}{\beta + V_l(X) + V_l(Y)} \quad (5.20)$$

where $V_l(X)$ and $V_l(Y)$ are the shape complexity of X and Y at ℓ^{th} level and $\beta=0.35$ is constant. $V_l(X)$ is defined as

$$V_l(X) = \frac{1}{n'_2 + n'_1} \sum_{t=1}^{n'_2+n'_1} std(FNDES_l^{x_1}(t), \dots, FNDES_l^{x_N}(t)) \quad (5.21)$$

where $std(\cdot)$ denotes the standard deviation.

To improve retrieval accuracy, we use another cost function, $R(X, Y)$. Let A , B , and C be images that have the same object but have shape variations. Assume that A is similar to B , B is similar to C , but A is not similar to C . Using the similarity between A and B with the similarity between B and C is helpful to match A and C . This idea was also addressed in [174] but it requires pre-computed similarities between all pairs of images and a complex learning process. In this thesis we describe a simpler method. First, we find the top M images that minimize $C_M(X, Y)$ from the database. Let the top M image be X_{sim}^k for $k = 1, 2, \dots, M$. Then, $C_M(X_{sim}^k, Y)$ is computed and summed for all $k = 1, 2, \dots, M$ as

$$R(X, Y) = \sum_{k=1}^M C_M(X_{sim}^k, Y) \quad (5.22)$$

where $R(X, Y)$ is the sum of cost between X_{sim}^k and Y . We call this matching a inductive matching. Our final cost between X and Y is defined as

$$C_T(X, Y) = C_M(X, Y) + R(X, Y) \quad (5.23)$$

The most N_{sim} similar database images are retrieved by minimizing $C_T(X, Y)$.

5.3 Experimental Results

In our experiments we used two publicly available datasets. Dataset 1 is the MPEG-7 shape dataset [175] and Dataset 2 is the Articulated dataset [164]. Figure

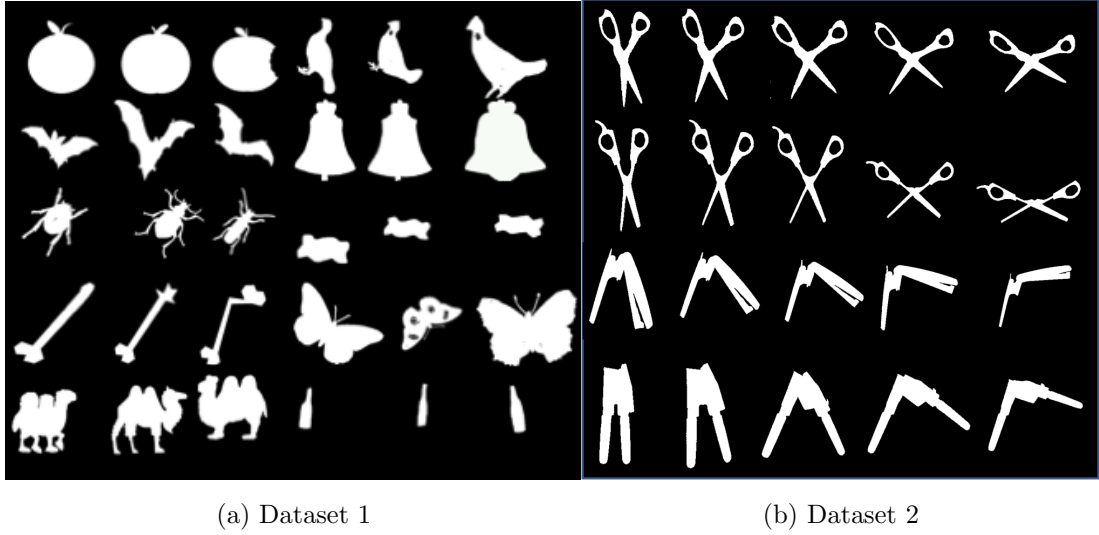


Fig. 5.2.: Sample shape images in our database

5.2 shows some sample images from our database. In the experiment the number of contour points, $N=200$ is used for both datasets. Also, the number of neighbor points at the lowest level for multiple SSAI descriptors, $N_{lowest}=20$ is used for both datasets.

The MPEG-7 dataset consists of 1400 silhouette images with 70 types of objects, each having 20 different shapes [175]. As shown in Figure 5.2a the images in this dataset include many non-rigid deformations. To evaluate the performance of retrieval methods we used the “bull’s eyes score” [175]. For this score, each image is used as an input image and we count how many images are correctly matched to the input image within the top 40 rank images. We then compute the sum of the number of correctly matched images for all the images in the dataset. This summation is divided by 20×1400 to form the bull’s eyes score. A “perfect” bull’s eyes score is 100. For this dataset our method is compared with several other existing methods. We set $M = 9$ and $L = 8$ where M is the number of images used for inductive matching and L is the number of levels used for multiple SSAI descriptors. Table 5.1 shows the experimental results. Our method based on a single SSAI descriptor

with $\ell = 1$ and $\ell = 4$ achieved 75.66 and 82.02 respectively, which are better than many of the existing methods. This shows the shape images in this dataset can be represented using SSAI descriptor and they can be recognized better using a single SSAI descriptor at higher level. The use of multiple SSAI descriptors achieved 87.73, which is better than several existing methods including IDSC [164], HPM [165] and Shape tree [166]. This shows multiple SSAI descriptors give better accuracy to the shape images that include non-rigid deformations. Multiple SSAI descriptors with inductive matching achieved 90.60, which is more accurate than all other existing methods except LP [174]. However, our inductive matching is much simpler than LP because our method does not need to pre-compute the similarities between all pairs of images or require a learning process.

The Articulated dataset consists of 40 silhouette images with 8 types of objects, each having 5 different shapes. To evaluate the performance of retrieval methods we show the number of top 1 to top 4 closest matches for the same object [164]. The best possible result for the ranking is 40. For this dataset, $M = 1$ and $L = 1$ give the best result. Table 5.2 show the experimental results. Our methods are more accurate than all the existing methods. Note that $M = 1$ means that multiple SSAI is the same as single SSAI at $\ell = 1$. The reason why single SSAI is better than multiple SSAI in this dataset is that this dataset includes several different object images that look similar. As shown in Figure 5.2b, images in the first row are different objects from images in the second row, but they look similar. The small differences between the images in the first row and the second row cannot be distinguished by the smoothed SSAI descriptors at higher level.

5.4 Conclusions and Future Work

We described a local affine invariant shape descriptor, SSAI, based on self similarity of object shape that is invariant to affine transform. By using multiple SSAI descriptors based on different set of neighbor points, we recognize object shapes more

Table 5.1.: Bull’s Eye Score - MPEG-7 dataset

Method	Score (%)
CSS [162]	75.44
SC [117]	76.51
Generative model [169]	80.03
An ordered BOF [171]	80.07
IDSC [164]	85.40
Symbolic Representation [126]	85.92
HPM [165]	86.35
Shape tree [166]	87.70
Self closed partial [160]	88.26
Metric partition constraint [163]	88.90
Contour flexibility [167]	89.31
LAI [125]	89.62
Height function + shape complexity [161]	90.35
LP [174]	91.00
Our Method (Single SSAI at $\ell=1$)	75.66
Our Method (Single SSAI at $\ell=4$)	82.02
Our Method (Multi SSAI)	87.73
Our Method (Multi SSAI + Inductive Matching)	90.60

Table 5.2.: Retrieval result - Articulated dataset

Method	Top 1	Top 2	Top 3	Top 4
SC [117]	20/40	10/40	11/40	5/40
IDSC [164]	40/40	34/40	35/40	27/40
Metric partition constraint [163]	39/40	32/40	18/40	16/40
LAI [125]	39/40	39/40	37/40	21/40
Our Method (Multi SSAI)	40/40	40/40	39/40	34/40
Our Method (Multi SSAI + Inductive Matching)	40/40	40/40	40/40	40/40

accurately under non-rigid deformation. Our image matching combined with inductive matching helps to recognize object shape more accurately. In this thesis we used a fixed number of levels, M for multiple SSAI descriptors for all the input images, but the best value of M that recognizes objects with the highest accuracy can be different depending on input image. For the future work, we will investigate the image matching method to choose the best value of M adaptively. Also, we will improve our inductive matching using more similarities between images.

6. SYSTEM IMPLEMENTATION

6.1 Overall System

The goal of this system is to develop integrated mobile-based systems capable of tattoo image analysis. This system will provide accurate and useful information to the law enforcement officers based on a database of gang tattoo images in real time. Using a mobile application of our system, a user can take a picture of tattoo and send to an image analysis server. Then, the image will be analyzed and compared to the other images in a database. The image analysis results will be sent back to a user, and the user can see the results.

We implemented a tattoo image analysis system as an application for iPhone and Android devices and as a web-based interface accessible from any web browser. Our tattoo analysis system is a part of the GARI gang graffiti system [124, 176, 177]. Our tattoo image analysis system has the same features as GARI for graffiti image analysis but uses different segmentation and image retrieval methods. Our tattoo image analysis system is illustrated in Figure 6.1.

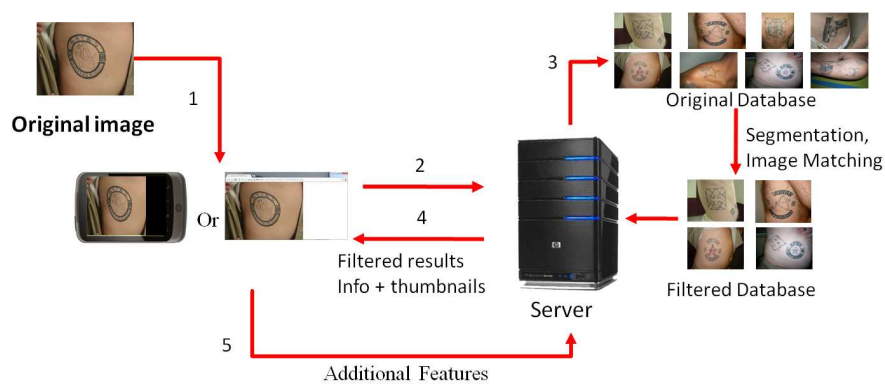


Fig. 6.1.: Overview of the Tattoo Image Analysis System

6.2 Mobile Application

To enter the GARI system, a user must input his user ID and a unique password assigned to each user on the login screen. It is depicted in Figure 6.2.

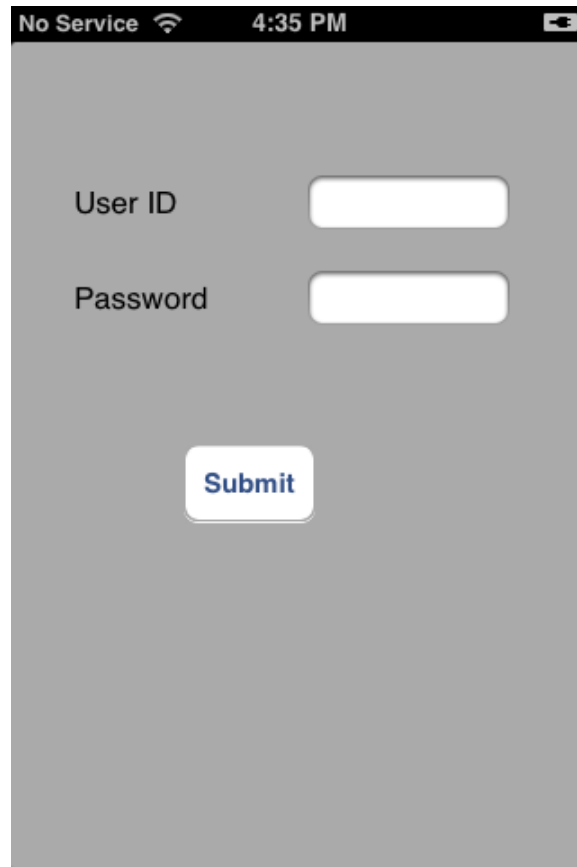


Fig. 6.2.: An Example of a Login

Once the User ID and password has been entered, the main screen is presented as shown in Figure 6.3(a). After an image is captured or browsed, the secondary screen (Figure 6.3(b)) is shown. These main screens include the following options:

- Browse Image
- Browse Database
- Capture Image

- Send to Server (available after browsing or capturing an image)
- Settings
- About

We introduce the options related to tattoo images in more detail.

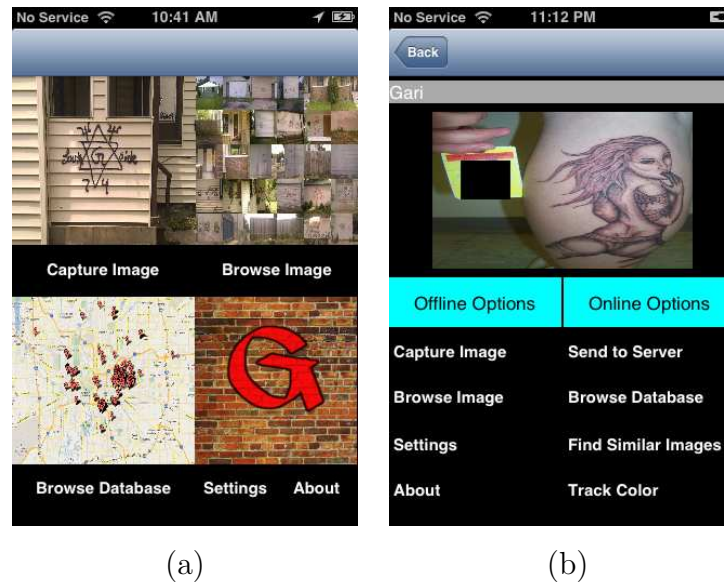


Fig. 6.3.: User Options: (a) is the main screen and (b) is the secondary screen

6.2.1 Browse Image

The users can browse images they acquired and stored on the iPhone device. They can upload the images to the server or analyze them later. The examples of this is shown in Figure 6.4.

6.2.2 Browse Database

The menu option “Browse Database” allows the user to browse the graffiti and tattoo database by radius, date and gang name. When a user clicks the button, “Browse

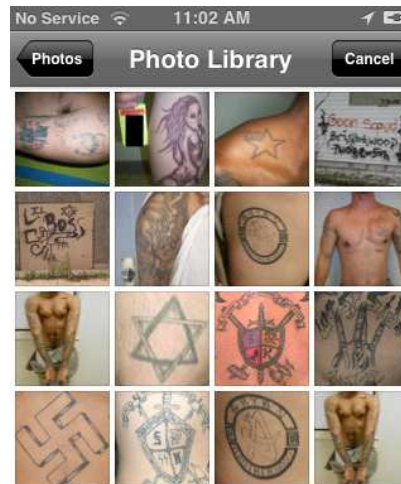


Fig. 6.4.: An Example of Image Browsing

Database,” the user can choose one option from 4 different options as shown in Figure 6.5.

Browse Database by Radius

Using this option, a user can view all the images from the database within a given radius from the current location. Examples of this are depicted in Figure 6.6. Since most tattoo images do not have geolocation information, the browsed database results by radius are graffiti images.

Browse Database by Date

Using this option, a user can view all the images from the database uploaded within a time interval. The examples of this are shown in Figure 6.7. The browsed

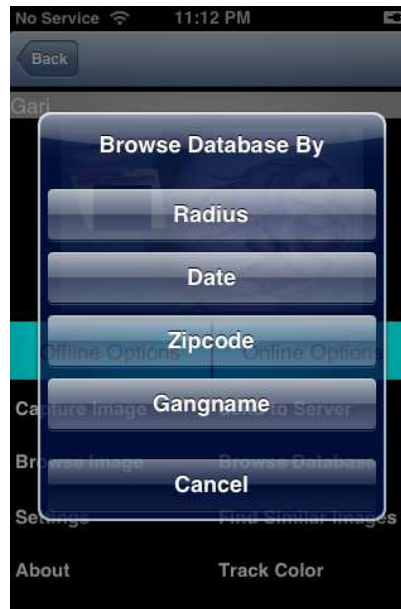
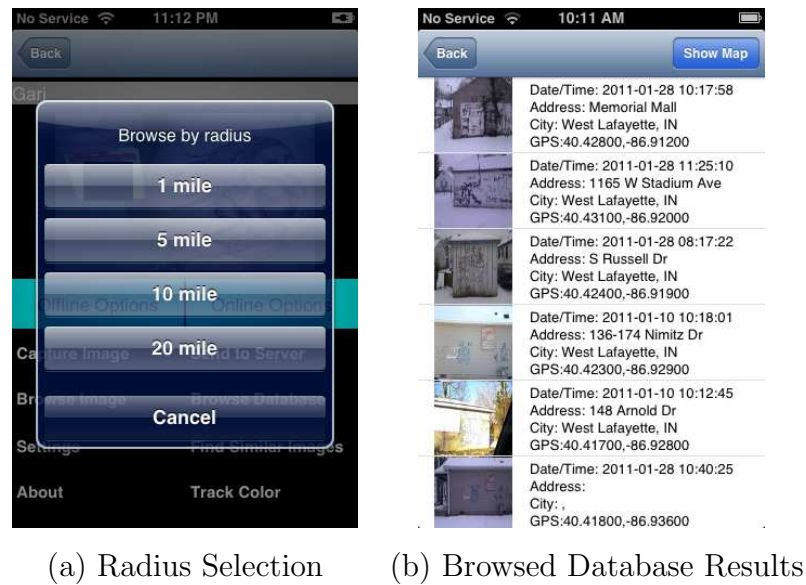


Fig. 6.5.: Four Different Options in Browse Database



(a) Radius Selection (b) Browsed Database Results

Fig. 6.6.: Examples of Browsing Database By Radius

database results show graffiti images as well as tattoo images uploaded between start date and end date.

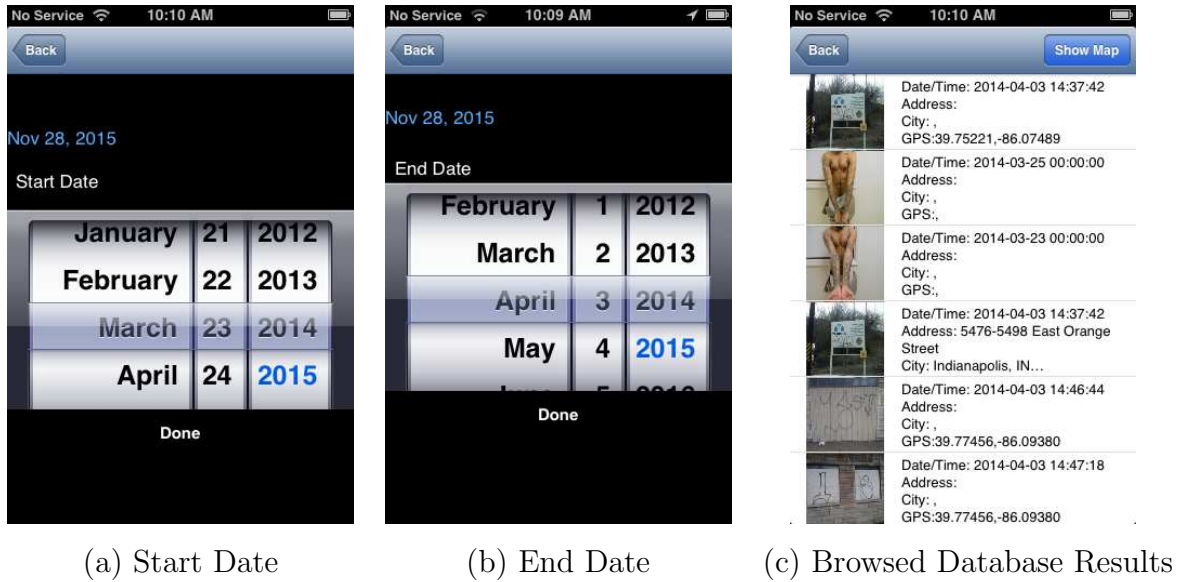


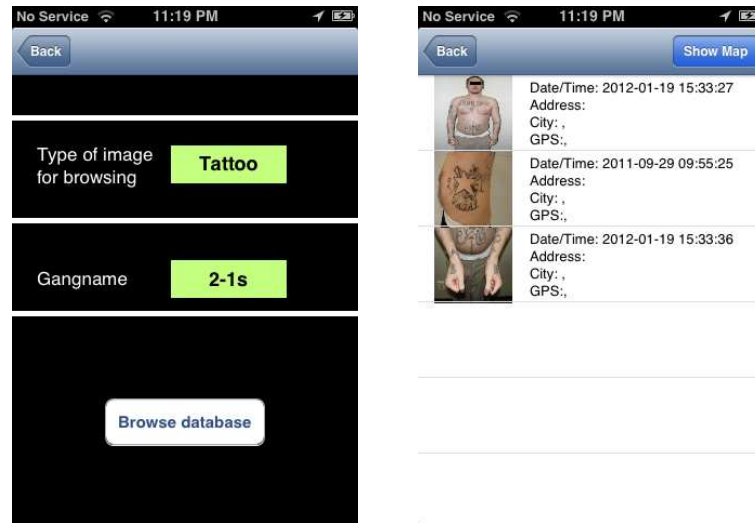
Fig. 6.7.: Examples of Browsing Database By Date

Browse Database by Gang Name

Using this option, a user can view all the images from the database that have the gang name the user specifies. The examples of this are depicted in Figure 6.8. In the example we choose the type of an image as tattoo and gang name as “2-1S”, so three tattoo images related to the gang name are retrieved.

6.2.3 Capture Image

The user can take a picture of a tattoo image by clicking the button “Capture Image.” Examples of this are shown in Figure 6.9. Note that the user’s current location is also saved after acquiring an image.



(a) Gang Name Selection (b) Browsed Database Results

Fig. 6.8.: Examples of Browsing Database By Gang Name

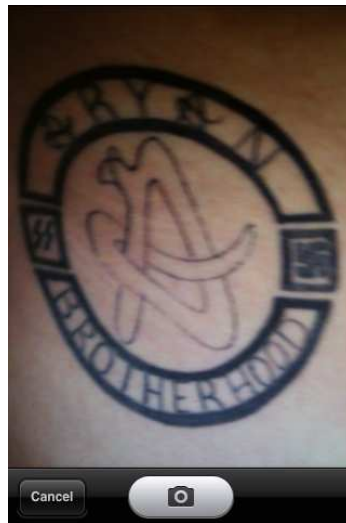
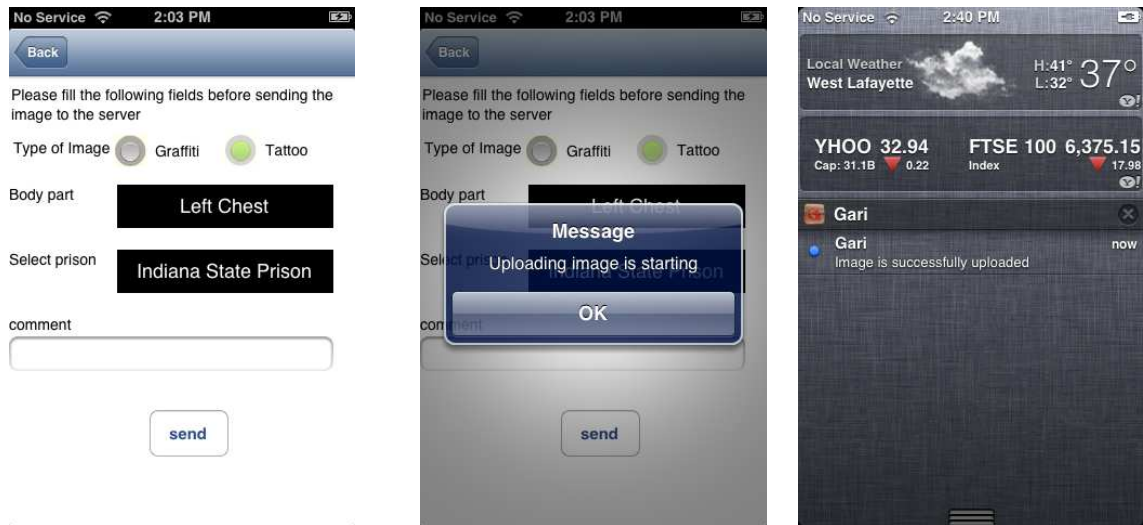


Fig. 6.9.: An Example of Image Capture

6.2.4 Send to Server

Our “Send to Server” option allows users to send the images they acquired to the GARI server. Once the users send the images, the communication process is done as

a background process, so the users can keep sending images without waiting for the response from the server. The examples of this are depicted in Figure 6.10.



(a) A Form for Sending Image (b) Start to Send Image (c) Finish Sending Image

Fig. 6.10.: Examples of Sending Images to Server

6.2.5 Find Similar Images

The “Find Similar Images” option allows users to find images similar to the current image displayed on the secondary screen. When a user clicks the “Find Similar Images” button, the user can choose the type of an image between graffiti and tattoo. Once the “tattoo” button is clicked, the image is sent to the server and analyzed based on our tattoo image segmentation and tattoo image matching methods. When the analysis is done, the server sends back a list of matching candidates. Note that the matching candidates are sorted by the order of the most similar image. Examples of this are shown in Figure 6.11.

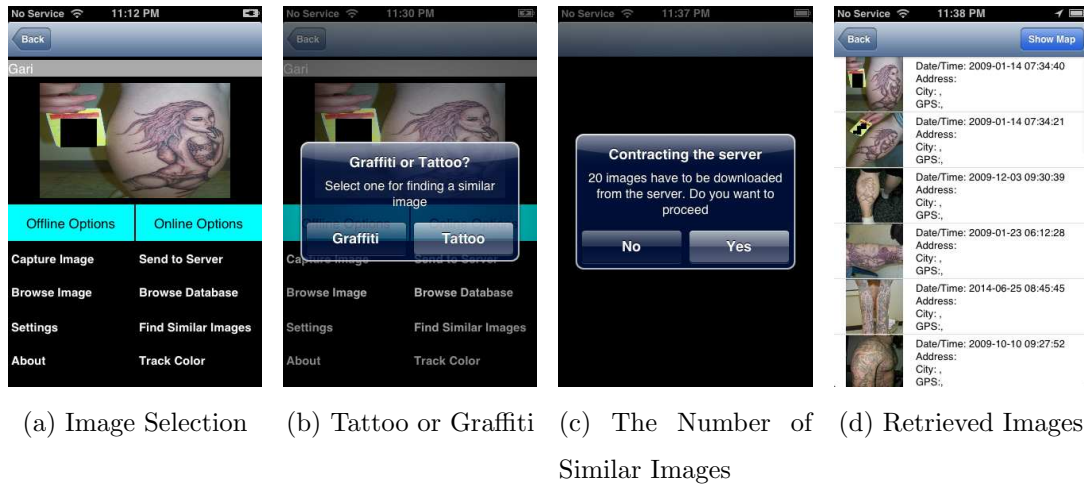


Fig. 6.11.: Examples of Finding Similar Images

6.2.6 Setting

All authorized users can use the “Setting” option. This feature has various options.

- Server domain/IP : the address of the server can be changed by domain name or IP address
- Switch user : another user can login while the application is running
- Change password : a user can change his login password while the application is running
- Send crashlog : a user can send crash feedback to the developer using this option

Examples of this is shown in Fig. 6.12.

6.3 Web Interface System

Our system also supports a web interface. Since our system uses geolocation, which is introduced with HTML5, our web interface can be used in any web browser that supports HTML5. To enter the GARI system, a user must input his user ID

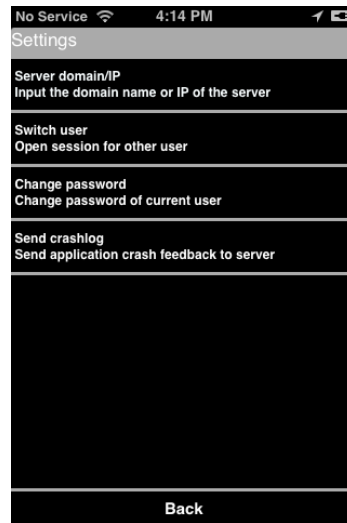


Fig. 6.12.: An Example of the Setting Option

and a unique password assigned to each user on the login screen as same as mobile version. It is depicted in Figure 6.13.

Fig. 6.13.: An Example of Login

Once the User ID and password has been entered, the main menu is presented as depicted in Figure 6.14.

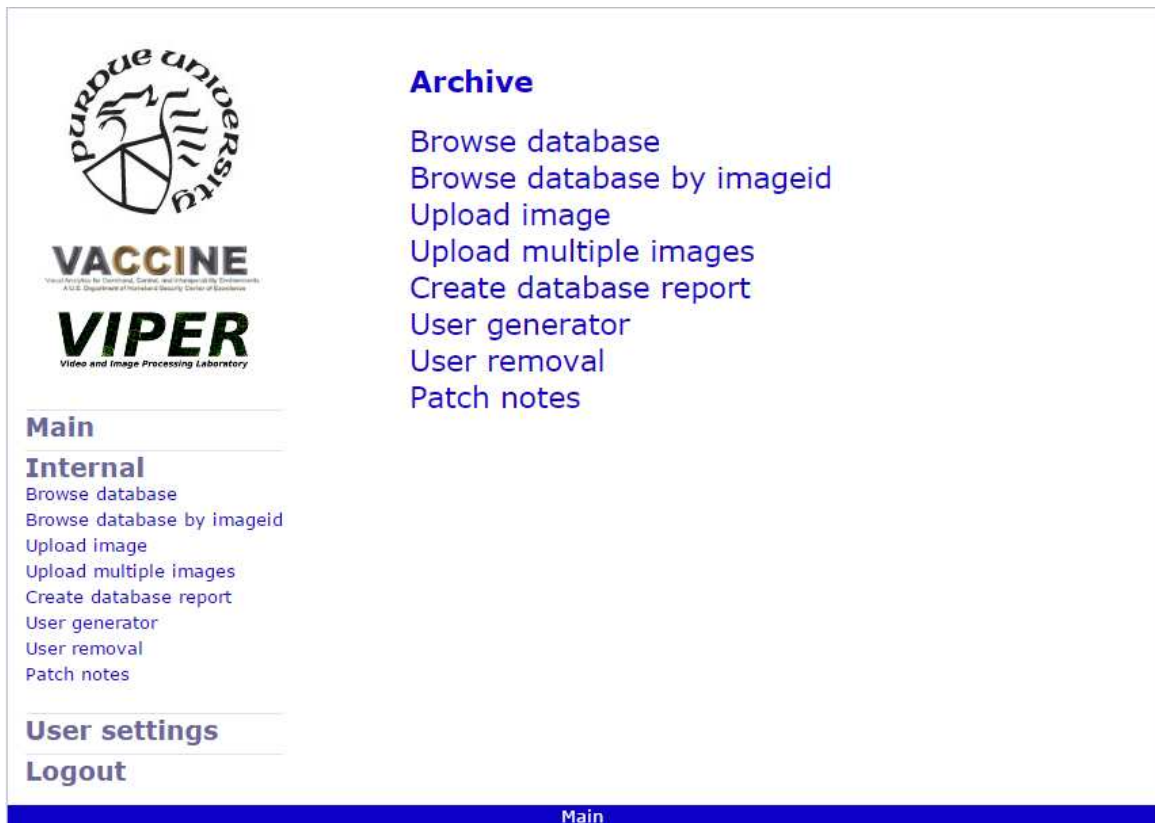


Fig. 6.14.: Main Menu in Web Interface

This main menu includes the following options:

- Browse Database
- Browse Database by imageid
- Upload Image
- Upload multiple Image
- Create database report

- User generator
- User removal
- Patch notes

Since “Browse Database” option is only related to tattoos, we will describe it in more detail.

6.3.1 Browse Database

In the “Browse Database” a user can browse our tattoo database by date, first responder ID, and gang name.

Browse Database by Date

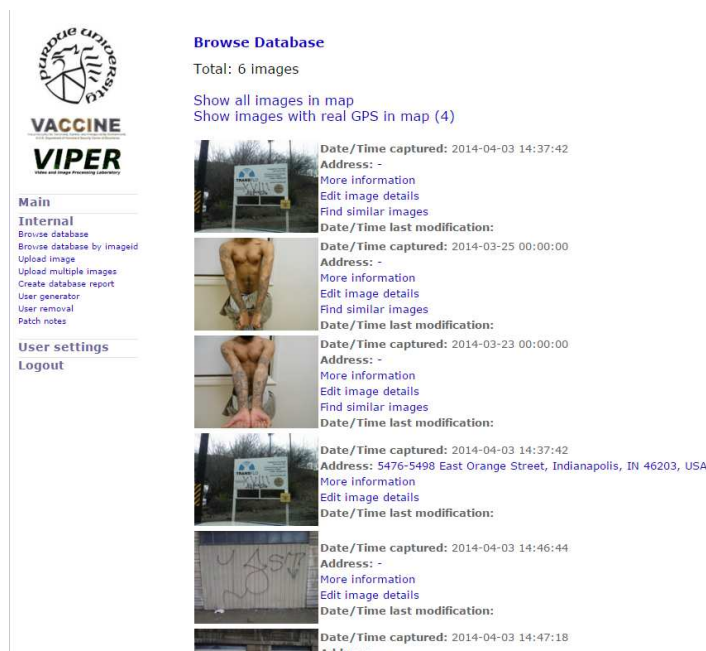
To use this feature, a user should specify the time interval when tattoo image are uploaded, captured, or modified. A user then can retrieve all the tattoo images of the time interval by clicking a button, “tattoo”. An example of this is shown in Figure 6.15.

Browse Database by Gangname

Using this option, a user can retrieve all the images from the database that have the same gang name chosen. An Example of this is shown in Figure 6.16.

Browse Database by First Responder ID

Using this option, a user can retrieve all the images from the our database that the person of the given first responder ID took. The example of this is depicted in Figure 6.17.



Browse Database
Total: 6 images

Show all images in map
Show images with real GPS in map (4)








	Date/Time captured: 2014-04-03 14:37:42 Address: - More information Edit image details Find similar images Date/Time last modification:
	Date/Time captured: 2014-03-25 00:00:00 Address: - More information Edit image details Find similar images Date/Time last modification:
	Date/Time captured: 2014-03-23 00:00:00 Address: - More information Edit image details Find similar images Date/Time last modification:
	Date/Time captured: 2014-04-03 14:37:42 Address: 5476-5498 East Orange Street, Indianapolis, IN 46203, USA More information Edit image details Date/Time last modification:
	Date/Time captured: 2014-04-03 14:46:44 Address: - More information Edit image details Date/Time last modification:
	Date/Time captured: 2014-04-03 14:47:18 Address: -

Fig. 6.15.: An Example of Browse Database by Date

6.3.2 Find Similar Images

The “Find Similar Images” option allows a user to find similar images. In the list of images from Browse Database, a user can click the button, “Find similar images”. If the type of the image the user choose is a tattoo, then the most similar 20 images are retrieved using our tattoo image matching method. An example of it is shown in Figure 6.18.



VACCINE
VIPER
video and image processing laboratory

Main

Internal

- Browse database
- Browse database by imageid
- Upload image
- Upload multiple images
- Create database report
- User generator
- User removal
- Patch notes

User settings

Logout

Browse Database

Total: 28 images

Show all images in map
Show images with real GPS in map (0)






	<p>Date/Time captured: 2014-09-18 17:55:55 Address: - More information Edit image details Find similar images Gangname selected: Aryan Brotherhood, unknown, unknown, unknown, unknown</p>
	<p>Date/Time captured: 2013-02-19 13:34:15 Address: - More information Edit image details Find similar images Gangname selected: Aryan Brotherhood, unknown, unknown, unknown, unknown</p>
	<p>Date/Time captured: 2014-09-18 17:56:16 Address: - More information Edit image details Find similar images Gangname selected: Aryan Brotherhood, unknown, unknown, unknown, unknown</p>
	<p>Date/Time captured: 2014-09-18 17:56:51 Address: - More information Edit image details Find similar images Gangname selected: Aryan Brotherhood, unknown, unknown, unknown, unknown</p>

Fig. 6.16.: An Example of Browse Database by Gang Name



VACCINE
Video and Image Processing Laboratory

VIPER

Main

Internal

- Browse database
- Browse database by imageid
- Upload image
- Upload multiple images
- Create database report
- User generator
- User removal
- Patch notes


User settings

Logout


Browse Database

Total: 79 Images


[Show all images in map](#)
[Show images with real GPS in map \(0\)](#)




Date/Time captured: 2014-09-18 18:36:56
Address: -
[More information](#)
[Edit image details](#)
[Find similar images](#)



Date/Time captured: 2014-09-18 18:23:57
Address: -
[More information](#)
[Edit image details](#)
[Find similar images](#)



Date/Time captured: 2014-09-18 18:36:03
Address: -
[More information](#)
[Edit image details](#)



Date/Time captured: 2014-09-18 17:50:44
Address: -
[More information](#)
[Edit image details](#)

Fig. 6.17.: An Example of Browse Database by First Responder ID

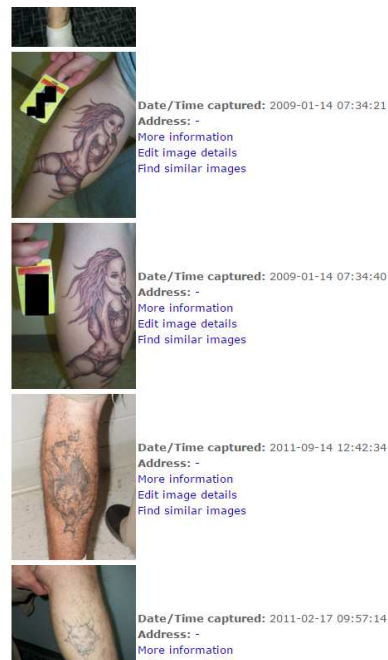


Fig. 6.18.: An Example of Finding Similar Images

7. CONCLUSIONS

7.1 Summary

In this thesis we first investigate a tattoo segmentation method that removes background clutter from an image. We then introduce our tattoo image retrieval system based on the proposed local and global image descriptors. We also improve our previous image descriptor and image matching method for more accurate image retrieval. We describe our submissions to the Tatt-C Tattoo Identification (TID) and the Tatt-C Region of Interest (ROI). We propose our spatial pyramid alignment technique for sparse coding based object classification. This technique is used on the tattoo image dataset as well as public object recognition image datasets. Last, we introduce a shape descriptor known as Self Similar Affine Invariant (SSAI) descriptor for shape retrieval. The main contributions of this thesis are listed as follows:

- Efficient Graph-Cut Tattoo Segmentation

We propose a graph-cut tattoo segmentation method based on image edges, a skin color model, and a visual saliency map to find skin pixels around tattoo regions. The post processing, that detects the skin pixels around a tattoo only from the graph-cut segmentation results, is then used. The method was evaluated on datasets that were collected from the Indiana State Police, eviltattoo.com [61], and NIST tattoo challenge dataset [13]. Experimental evaluation demonstrates that our segmentation method can detect and segment tattoo regions correctly even when a tattoo image includes background clutter.

- Efficient Graph-Cut Tattoo Segmentation with Body Boundary Removal

We propose body boundary removal (BBR) method to improve our previous segmentation method. The previous method makes errors when tattoo regions

are very close to the boundaries of human body. Thus, our BBR method fixes the errors by removing the skin pixels on the boundaries of human body. Experimental results demonstrate that the segmentation errors of our previous method are reduced using BBR.

- Tattoo Image Matching and Retrieval Based on Local Image Descriptor (MHLC Descriptor) and Global Image Descriptor Robust to Image Deformations

We create new local and global shape descriptors robust to scale, translation, rotation, and shape distortions for tattoo image retrieval. By using the scale invariant feature transform (SIFT) with local shape context based on multiple different sized-bin polar histograms (MH), more accurate image matching can be obtained. A global shape descriptor based on MH and a 2D Fourier Transform is also used for robustness of translation, scale, rotation and shape distortions. We also describe robust similarity for local descriptors and a weighted matching method based on local and global descriptors. Experimental results show that our method outperforms several existing methods.

- Tattoo Image Matching and Retrieval Based on Modified MHLC Descriptor (DMHLC Descriptor)

We introduce the improved MHLC descriptor, called as DMHLC descriptor. Instead of using the spatial distribution of the SIFT features, the DMHLC descriptor uses the spatial distribution of the densely sampled features on the tattoo object to generate the multiple polar histograms. The multiple polar histograms are combined with the SIFT descriptor to generate DMHLC descriptor. Our experimental results show that our DMHLC descriptor improves the image retrieval accuracy much more than our MHLC descriptor.

- Modified Inductive Matching

We introduce our modified inductive matching to improve the image retrieval accuracy. By considering all the similarities between all the images in the

database, the image retrieval accuracy is improved. The modified inductive matching retrieves the most dissimilar M_2 database images respect to an input image first. Then, the mean of the image similarities between the M_2 database images and one database image is considered to compute the final image similarity between the input image and the database image. Our experimental results show that the modified inductive matching improves the image retrieval accuracy more than the pairwise image similarity based on the image matching of two images.

- The Our Submissions to NIST Tattoo Recognition Technology Challenge

For tattoo image retrieval on NIST Tattoo Identification (TID) dataset, the tattoo image retrieval system based on the MHLC descriptor is introduced. Experimental results show that our method outperforms the method of [4]. Our method also outperforms five different methods reported in the NIST challenge [11]. For tattoo image retrieval on NIST Region of Interest (ROI) dataset, we also create another image descriptor based on local self similarity (LSS) [62] and SIFT. We also propose a weighted distance similarity metric to retrieve the most similar images from the test dataset. Experimental results demonstrate that our method outperforms the method of [4]. Our method also outperforms four different methods reported in the NIST challenge [11].

- Spatial Pyramid Alignment For Sparse Coding Based Object Classification

We propose a simple but efficient spatial pyramid alignment method that can be combined with the existing sparse coding methods. By using max pooled features, we estimate an object center and align the spatial pyramid accordingly. We also propose an image representation descriptor robust to misalignment and object deformations using max pooling on multiple image descriptors generated by shifting the pyramid center in a pre-defined margin. We test the modified center-aligned spatial pyramid with the sparse coding method on the tattoo image dataset as well as public object recognition image datasets. Our experi-

mental results show that our proposed spatial pyramid with the sparse coding improve the image object classification accuracy more than the original spatial pyramid with the same sparse coding.

- Shape Matching Using A Self Similar Affine Invariant Descriptor

We introduce a shape descriptor known as Self Similar Affine Invariant (SSAI) descriptor for shape retrieval. The SSAI descriptor is based on the property that two sets of points are transformed by an affine transform, then subsets of each set of points are also related by the same affine transformation. Also, the SSAI descriptor is insensitive to local shape distortions. We use multiple SSAI descriptors based on different sets of neighbor points to improve shape recognition accuracy. We also describe an efficient image matching method for the multiple SSAI descriptors. Experimental results show that our approach achieves very good performance on two publicly available shape datasets.

7.2 Future Work

Our methods can be improved and extended in the following ways:

- Tattoo Segmentation

We reviewed current tattoo segmentation and localization techniques and presented a tattoo segmentation based on image edges, a skin color model, a visual saliency map, and body boundary removal. Our method still makes segmentation errors when there are hair regions with strong edges inside a body. In future, we plan to improve the post processing in our segmentation by finding the region whose shape looks like a tattoo.

- Tattoo Image Retrieval based on Image Matching

We reviewed the major techniques to the tattoo image retrieval and described several different tattoo image retrieval systems based on three new local image descriptors (MHLC descriptor, DMHLC descriptor and LSS+SIFT descriptor)

and one global image descriptor. We also described the modified inductive matching and the weighted distance image similarity (WDS) to improve the image retrieval accuracy. Currently our image retrieval speed is slow because an input image should be compared with all the images in the database to retrieve the most similar N images. In future, we plan to investigate the methods to make our retrieval process fast by using tree structures for retrieval process. Also, deep neural network learning based image retrieval method will be consider to improve the image retrieval accuracy.

- **Tattoo Image Classification Based On Sparse Coding**

We reviewed the existing sparse coding with spatial pyramid methods for object classification and presented the spatial pyramid alignment method that can be combined with any sparse coding method. This method achieved good classification accuracies even when there are the object shape distortion and an image translation. However, our spatial pyramid alignment method does not work well when there are lots of background clutters in an image. For the future work, we will investigate the method to estimate an object center correctly even when there is background clutter.

7.3 Publications Resulting From Our Work

Journal Papers

1. **J. Kim** and E. J. Delp, “Tattoo Image Retrieval Based On Robust Tattoo Image Matching”, *To be submitted to the IEEE Transactions on Information Forensics and Security*.
2. **J. Kim**, K. Tahboub, and E. J. Delp, “Shape Matching and Retrieval Using a Self Similar Affine Invariant Descriptor”, *To be submitted to the IEEE Signal Processing Letters*.

Conference Papers

1. **J. Kim**, K. Tahboub, and E. J. Delp, "Spatial Pyramid Alignment For Sparse Coding Based Object Classification", to appear *Proceedings of the IEEE International Conference on Image Processing*, Beijing, China.
2. **J. Kim**, H. Li, J. Yue, and E. J. Delp, "Shape Matching Using a Self Similar Affine Invariant Descriptor", *Proceedings of the IEEE International Conference on Image Processing*, pp. 2470-2474, September, 2016, Phoenix, AZ.
3. **J. Kim**, H. Li, J. Yue, J. Ribera, L. Huffman, and E. J. Delp, "Automatic and Manual Tattoo Localization", *Proceedings of the IEEE International Conference on Technologies for Homeland Security*, pp. 1-6, May 2016, Waltham, MA.
4. **J. Kim**, H. Li, J. Yue, and E. J. Delp, "Tattoo Image Retrieval for Region of Interest", *Proceedings of the IEEE International Conference on Technologies for Homeland Security*, pp. 1-6, May 2016, Waltham, MA.
5. **J. Kim**, A. Parra, J. Yue, H. Li, and E. J. Delp, "Robust Local and Global Shape Context for Tattoo Image Matching", *Proceedings of the IEEE International Conference on Image Processing*, pp. 2194-2198, October 2015, Quebec, Canada.
6. **J. Kim**, A. Parra, H. Li, and E. J. Delp, "Efficient Graph-Cut Tattoo Segmentation", *Proceedings of the SPIE/IS&T Conference on Visual Information Processing and Communication VI*, pp. 94100H-1-8, February 2015, San Francisco, CA.
7. A. Parra , B. Zhao, **J. Kim**, and E. J. Delp, "Recognition, Segmentation and Retrieval of Gang Graffiti Images on a Mobile Device", *Proceedings of the IEEE International Conference on Technologies for Homeland Security*, pp. 178-183, November 2013, Waltham, MA.

REFERENCES

REFERENCES

- [1] P. Duangphasuk and W. Kurutach, "Tattoo skin detection and segmentation using image negative method," *Proceedings of the IEEE International Symposium on Communications and Information Technologies*, pp. 354–359, September 2013, Surat Thani, Thailand.
- [2] B. Heflin, W. Scheirer, and T. Boulton, "Detecting and classifying scars, marks, and tattoos found in the wild," *Proceedings of the IEEE International Conference on Biometrics: Theory, Applications and Systems*, pp. 31–38, September 2012, Arlington, VA.
- [3] D. Manger, "Large-scale tattoo image retrieval," *Proceedings of the IEEE Conference on Computer and Robot Vision*, pp. 454–459, May 2012, Toronto, ON.
- [4] J. Lee, R. Jin, A. Jain, and W. Tong, "Image retrieval in forensics: Tattoo image database application," *IEEE Transactions on Multimedia*, vol. 19, no. 1, pp. 40–49, January 2012.
- [5] J. Allen, N. Zhao, J. Yuan, and X. Liu, "Unsupervised tattoo segmentation combining bottom-up and top-down cues," *Proceedings of the SPIE Mobile Multimedia/Image Processing, Security, and Applications*, vol. 8063, pp. 80 630L–1–9, April 2011, Orlando, FL.
- [6] A. Jain, J. Lee, R. Jin, and N. Gregg, "Content-based image retrieval: An application to tattoo images," *Proceedings of the IEEE International Conference on Image Processing*, pp. 2745–2748, November 2009, Cairo, Egypt.
- [7] J. Lee, R. Jin, and A. Jain, "Rank-based distance metric learning: An application to image retrieval," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2008, Anchorage, AK.
- [8] S. Acton and A. Rossi, "Matching and retrieval of tattoo images: Active contour CBIR and glocal image features," *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 21–24, March 2008, Santa Fe, NM.
- [9] A. Jain, J. Lee, and R. Jin, "Tattoo-ID: Automatic tattoo image retrieval for suspect and victim identification," *Proceedings of the Pacific Rim Conference on Multimedia*, pp. 256–265, December 2007, Hong Kong, China.
- [10] A. Jain, R. Jin, and J. Lee, "Tattoo image matching and retrieval," *IEEE Computer*, vol. 45, no. 5, pp. 93–96, May 2012.
- [11] M. Ngan, G. Quinn, and P. Grother, "Tattoo recognition technology challenge (Tatt-C) outcomes and recommendations," *NIST Internal Report*, September 2015, National Institute of Standards and Technology, Gaithersburg, MD. [Online]. Available: <http://dx.doi.org/10.6028/NIST.IR.8078>

- [12] “Tatt-C,” URL:<http://www.nist.gov/itl/iad/ig/tatt-c.cfm>.
- [13] M. Ngan and P. Grother, “Tattoo recognition technology - challenge (Tatt-C): An open tattoo database for developing tattoo recognition research,” *Proceedings of the IEEE International Conference on Identity, Security and Behavior Analysis*, pp. 1–6, March 2015, Hong Kong, China.
- [14] D. Marcetic, S. Ribaric, V. Struc, and N. Pavesic, “An experimental tattoo de-identification system for privacy protection in still images,” *Proceedings of the IEEE International Convention on Information and Communication Technology, Electronics and Microelectronics*, pp. 1288–1293, May 2014, Opatija, Croatia.
- [15] P. Duangphasuk and W. Kurutach, “Tattoo skin cross - correlation neural network,” *Proceedings of the IEEE International Symposium on Communications and Information Technologies*, pp. 489–493, September 2014, Incheon, South Korea.
- [16] T. Hrka, K. Brki, S. Ribari, and D. Mareti, “Deep learning architectures for tattoo detection and de-identification,” *Proceedings of the IEEE International Workshop on Sensing, Processing and Learning for Intelligent Machines*, pp. 1–5, July 2016, Aalborg, Denmark.
- [17] Z. H. Sun, J. Baumes, P. Tunison, M. Turek, and A. Hoogs, “Tattoo detection and localization using region-based deep learning,” *Proceedings of the IEEE International Conference on Pattern Recognition*, pp. 3050–3055, December 2016, Cancun, Mexico.
- [18] B. Li and S. Acton, “Active contour external force using vector field convolution for image segmentation,” *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2096–2106, August 2007.
- [19] C. Rother, V. Kolmogorov, and A. Blake, “GrabCut: Interactive foreground extraction using iterated graph cuts,” *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 309–314, August 2004.
- [20] T. Boulton, R. Micheals, X. Gao, and M. Eckmann, “Into the woods: Visual surveillance of noncooperative and camouflaged targets in complex outdoor settings,” *Proceedings of the IEEE*, vol. 89, no. 10, pp. 1382–1402, August 2001.
- [21] D. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, November 2004.
- [22] V. Badrinarayanan, A. Kendall, and R. Cipolla, “SegNet: A deep convolutional encoder-decoder architecture for image segmentation,” *arXiv preprint arXiv:1511.00561*, November 2015.
- [23] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440, June 2015, Boston, MA.
- [24] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Region-based convolutional networks for accurate object detection and segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 1, pp. 142–158, January 2016.

- [25] A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian SegNet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding," *arXiv preprint arXiv:1511.02680*, November 2015.
- [26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Proceedings of Neural Information Processing Systems*, pp. 1097–1105, December 2012, Lake Tahoe, NV.
- [27] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *Proceedings of Neural Information Processing Systems*, pp. 91–99, December 2015, Montreal, Quebec, Canada.
- [28] J. Lee, A. Jain, and R. Jin, "Scars, marks and tattoos (SMT): Soft biometric for suspect and victim identification," *Proceedings of the IEEE Biometrics Symposium*, pp. 1–8, September 2008, Tampa, FL.
- [29] H. Han and A. Jain, "Tattoo based identification: Sketch to image matching," *Proceedings of the IEEE International Conference on Biometrics*, pp. 1–8, June 2013, Madrid, Spain.
- [30] X. Xu, M. Martin, and T. Bourlai, "Automatic tattoo image registration system," *Proceedings of the IEEE International Conference on Advances in Social Networks Analysis and Mining*, pp. 1238–1243, August 2016, San Francisco, CA.
- [31] L. Huffman and J. McDonald, "Mixed media tattoo image matching using transformed edge alignment," *Proceedings of the IEEE Symposium on Technologies for Homeland Security*, pp. 1–6, May 2016, Waltham, MA.
- [32] U. Park, J. Park, and A. K. Jain, "Robust keypoint detection using higher-order scale space derivatives: Application to image retrieval," *IEEE Signal Processing Letters*, vol. 21, no. 8, pp. 962–965, August 2014.
- [33] Q. Xu, S. Ghosh, X. Xu, Y. Huang, and A. W. K. Kong, "Tattoo detection based on CNN and remarks on the NIST database," *Proceedings of the IEEE International Conference on Biometrics*, pp. 1–7, June 2016, Aalborg, Denmark.
- [34] X. Di and V. M. Patel, "Deep tattoo recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 119–126, June 2016, Las Vegas, NV.
- [35] T. Pavlidis, "Limitations of content-based image retrieval," <http://www.theopavlidis.com/technology/CBIR/PaperB/vers3.htm>.
- [36] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," *Proceedings of the European Conference on Computer Vision Workshop on statistical Learning in Computer Vision*, pp. 59–74, 2004, Prague, Czech Republic.
- [37] F. Li, W. Tong, R. Jin, A. K. Jain, and J. Lee, "An efficient key point quantization algorithm for large scale image retrieval," *Proceedings of the ACM Workshop on Large-Scale Multimedia Retrieval and Mining*, pp. 89–96, October 2009, Beijing, China.

- [38] J. Lee, R. Jin, and A. Jain, “Unsupervised ensemble ranking: Application to large-scale image retrieval,” *Proceedings of the IEEE International Conference on Pattern Recognition*, pp. 3902–3906, September 2010, Orlando, FL.
- [39] S. Chopra, R. Hadsell, and Y. LeCun, “Learning a similarity metric discriminatively, with application to face verification,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 539–546, June 2005, San Diego, CA.
- [40] F. Wang, L. Kang, and Y. Li, “Sketch-based 3D shape retrieval using convolutional neural networks,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1875–1883, June 2015, Boston, MA.
- [41] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu, “Learning fine-grained image similarity with deep ranking,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1386–1393, June 2014, Columbus, OH.
- [42] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, “Selective search for object recognition,” *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, April 2013.
- [43] I. Endres and D. Hoiem, “Category independent object proposals,” *Proceedings of the European Conference on Computer Vision*, pp. 575–588, September 2010, Crete, Greece.
- [44] B. Alexe, T. Deselaers, and V. Ferrari, “Measuring the objectness of image windows,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2189–2202, November 2012.
- [45] J. Carreira and C. Sminchisescu, “CPMC: Automatic object segmentation using constrained parametric min-cuts,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1312–1328, July 2012.
- [46] F. Li and P. Perona, “A Bayesian hierarchical model for learning natural scene categories,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 524–531, June 2005, San Diego, CA.
- [47] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2169–2178, June 2006, New York, NY.
- [48] K. Grauman and T. Darrell, “The pyramid match kernel: discriminative classification with sets of image features,” *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1458–1465, October 2005, Cambridge, MA.
- [49] Z. Jiang, Z. Lin, and L. Davis, “Label consistent K-SVD: Learning a Discriminative dictionary for recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 11, pp. 2651–2664, November 2013.
- [50] A. Coates and A. Ng, “The importance of encoding versus training with sparse coding and vector quantization,” *Proceedings of the International Conference on Machine Learning*, pp. 921–928, June 2011, Bellevue, WA.

- [51] Q. Zhang and B. Li, “Discriminative K-SVD for dictionary learning in face recognition,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2691–2698, June 2010, San Francisco, CA.
- [52] J. Yang, K. Yu, Y. Gong, and T. Huang, “Linear spatial pyramid matching using sparse coding for image classification,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1794–1801, June 2009, Miami Beach, FL.
- [53] C. Weng, H. Wang, and J. Yuan, “Learning weighted geometric pooling for image classification,” *Proceedings of the IEEE International Conference on Image Processing*, pp. 3805–3809, September 2013, Melbourne, Australia.
- [54] J. Feng, B. Ni, Q. Tian, and S. Yan, “Geometric l_p -norm feature pooling for image classification,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2609–2704, June 2011, Colorado Springs, CO.
- [55] Y. Quan, Y. Xu, Y. Sun, Y. Huang, and H. Ji, “Sparse coding for classification via discrimination ensemble,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5839–5847, June 2016, Las Vegas, NV.
- [56] K. Yu, Y. Lin, and J. Lafferty, “Learning image representations from the pixel level via hierarchical sparse coding,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1713–1720, June 2011, Colorado Springs, CO.
- [57] L. Bo, X. Ren, and D. Fox, “Multipath sparse coding using hierarchical matching pursuit,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 660–667, June 2013, Portland, OR.
- [58] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, “Robust face recognition via sparse representation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, February 2009.
- [59] M. Aharon, M. Elad, and A. Bruckstein, “K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, November 2006.
- [60] M. Ranzato, Y. Boureau, and Y. Cun, “Sparse feature learning for deep belief networks,” *Proceedings of Neural Information Processing Systems*, pp. 1185–1192, December 2007, Vancouver, BC, Canada.
- [61] “eviltattoo.com,” URL:<http://eviltattoo.com>.
- [62] E. Shechtman and M. Irani, “Matching local self-similarities across images and videos,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2007, Minneapolis, MN.
- [63] J. Kim and K. Grauman, “Asymmetric region-to-image matching for comparing images with generic object categories,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2344–2351, June 2010, San Francisco, CA.
- [64] Y. J. Lee, J. Kim, and K. Grauman, “Key-segments for video object segmentation,” *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1995–2002, November 2011, Barcelona, Spain.

- [65] S. Kim, S. Ryu, B. Ham, J. Kim, and K. Sohn, "Local self-similarity frequency descriptor for multispectral feature matching," *Proceedings of the IEEE International Conference on Image Processing*, pp. 5746–5750, October 2014, Paris, France.
- [66] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, June 2014, Columbus, OH.
- [67] J. Carreira and C. Sminchisescu, "Constrained parametric min-cuts for automatic object segmentation," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3241–3248, June 2010, San Francisco, CA.
- [68] H. Yi, P. Yu, X. Xu, and A. W. K. Kong, "The impact of tattoo segmentation on the performance of tattoo matching," *Proceedings of the IEEE International WIE Conference on Electrical and Computer Engineering*, pp. 43–46, December 2015, IDhaka, Bangladesh.
- [69] Y. Li, J. Sun, C. Tang, and H. Shum, "Lazy snapping," *Proceeding of ACM SIGGRAPH*, pp. 303–308, August 2004, New York, NY.
- [70] 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, August 2000.
- [71] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266–277, August 2001.
- [72] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, January 1988.
- [73] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, November 2001.
- [74] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1124–1137, September 2004.
- [75] S. T. Acton, "Fast algorithms for area morphology," *Digital Signal Processing*, vol. 11, no. 3, pp. 187–203, July 2001.
- [76] J. Hartigan and M. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society, Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, August 1986.
- [77] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 8, pp. 679–698, August 1986.
- [78] A. Cheddad, J. Condell, K. Curran, and P. McKeivitt, "A skin tone detection algorithm for an adaptive approach to steganography," *Signal Processing*, vol. 89, no. 12, pp. 2465–2478, December 2009.

- [79] J. Kovac, P. Peer, and F. Solina, "Human skin colour clustering for face detection," *Proceedings of the IEEE Region 8 EUROCON 2003. Computer as a Tool*, pp. 144–148, September 2003, Ljubljana, Slovenia.
- [80] V. Nabiyev and A. Gunay, "Towards a biometric purpose image filter according to skin detection," *Proceedings of the Second international conference on problems of cybernetics and informatics*, pp. 1–4, September 2008, Baku, Azerbaijan.
- [81] K. Plataniotis and A. Venetsanopoulos, *Color Image Processing and applications*. New York: Springer, 2000.
- [82] K. Sobottka and I. Pitas, "A novel method for automatic face segmentation, facial feature extraction and tracking," *Signal Processing and Image Communication*, vol. 12, no. 3, pp. 263–281, 1998.
- [83] J. Tomaschitz and J. Facon, "Skin detection applied to multi-racial images," *Proceedings of the IEEE International Conference on Systems, Signals and Image Processing*, pp. 1–3, June 2009, Chalkida, Greece.
- [84] Y. Wang and B. Yuan, "A novel approach for human face detection from color images under complex background," *Journal of the Pattern Recognition Society*, vol. 34, no. 10, pp. 1983–1992, October 2001.
- [85] D. Chai and K. Ngan, "Face segmentation using skin-color map in videophone applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 4, pp. 551–564, June 1999.
- [86] M. R. Gupta and Y. Chen, "Theory and use of the EM algorithm," *Foundations and Trends in Signal Processing*, vol. 4, no. 3, pp. 223–296, April 2011.
- [87] Q. Zhu, C. Wu, K. Cheng, and Y. Wu, "A unified adaptive approach to accurate skin detection," *Proceedings of the IEEE International Conference on Image Processing*, pp. 1189–1192, October 2004, Singapore, Singapore.
- [88] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," *Proceedings of the Annual Conference on Neural Information Processing Systems*, pp. 545–552, December 2006, Vancouver, BC, Canada.
- [89] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, November 1998.
- [90] M. Jones and J. Rehg, "Statistical color models with application to skin detection," *International Journal of Computer Vision*, vol. 46, no. 1, pp. 81–96, January 2002.
- [91] J. Kim, A. Parra, H. Li, and E. J. Delp, "Efficient graph-cut tattoo segmentation," *Proceedings of the SPIE/IS&T Conference on Visual Information Processing and Communication VI*, vol. 9410, pp. 94100H–1–8, February 2015, San Francisco, CA.
- [92] J. Kim, H. Li, J. Yue, J. Ribera, E. J. Delp, and L. Huffman, "Automatic and manual tattoo localization," *Proceedings of the IEEE International Conference on Technologies for Homeland Security*, pp. 1–6, May 2016, Waltham, MA.

- [93] D. G. Lowe, "Object recognition from local scale-invariant features," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1150–1157, September 1999, Kerkyra, Greece.
- [94] C. Harris and M. Stephens, "A combined corner and edge detector," *Proceedings of the Alvey Vision Conference*, pp. 147–151, August 1988, Manchester, UK.
- [95] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, October 2004.
- [96] T. Tuytelaars and L. V. Gool, "Wide baseline stereo matching based on local, affinely invariant regions," *Proceedings of the British Machine Vision Conference*, pp. 38.1–38.14, September 2000, Bristol, UK.
- [97] J. Kim, C. Liu, F. Sha, and K. Grauman, "Deformable spatial pyramid matching for fast dense correspondences," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2307–2314, June 2013, Portland, OR.
- [98] O. Duchenne, A. Joulin, and J. Ponce, "A graph-matching kernel for object categorization," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1792–1799, November 2011, Barcelona, Spain.
- [99] 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*, vol. 32, no. 5, pp. 815–830, May 2010.
- [100] H. Bay, T. Tuytelaars, and L. Gool, "SURF: Speeded up robust features," *Proceedings of the European Conference on Computer Vision*, pp. 404–417, May 2006, Graz, Austria.
- [101] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," *Proceedings of the European Conference on Computer Vision*, pp. 481–495, September 2010, Crete, Greece.
- [102] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: Binary robust invariant scalable keypoints," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2548–2555, November 2011, Barcelona, Spain.
- [103] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, July 2002.
- [104] X. Li, M. Larson, and A. Hanjalic, "Pairwise geometric matching for large-scale object retrieval," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5153–5161, June 2015, Boston, MA.
- [105] Y. Hu and Y. Lin, "Progressive feature matching with alternate descriptor selection and correspondence enrichment," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 346–354, June 2016, Las Vegas, NV.

- [106] J. Huang, S. R. Kumar, M. Mitra, W. J. Zhu, and R. Zabih, "Image indexing using color correlograms," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 762–768, June 1997, San Juan, Puerto Rico.
- [107] A. K. Jain and A. Vailaya, "Shape-based retrieval: A case study with trademark image databases," *Pattern Recognition*, vol. 42, no. 9, pp. 1369–1390, September 1998.
- [108] T. Joachims, "Optimizing search engines using clickthrough data," *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 133–142, July 2002, Edmonton, AB, Canada.
- [109] H. Jegou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," *Proceedings of the European Conference on Computer Vision*, pp. 304–317, October 2008, Marseille, France.
- [110] M. Iturbe, O. Kähm, and R. Uribeetxeberria, "Surf and mu-surf descriptor comparison with application in soft-biometric tattoo matching applications," *Proceedings of the XII Spanish Meeting on Cryptology and Information Security*, pp. 345–349, September 2012, Donostia-San Sebastián, Spain.
- [111] M. Agrawal, K. Konolige, and M. Blas, "Censure: Center surround extremas for realtime feature detection and matching," *Proceedings of the European Conference on Computer Vision*, pp. 102–115, October 2008, Marseille, France.
- [112] M. J. Wilber, E. Rudd, B. Heflin, Y.-M. Lui, and T. E. Boulton, "Exemplar codes for facial attributes and tattoo recognition," *Proceedings of the Winter Conference on Applications of Computer Vision*, pp. 205–212, March 2014, Steamboat Springs, CO.
- [113] T. Malisiewicz, A. Gupta, B. Heflin, and A. Efros, "Ensemble of exemplar-svm for object detection and beyond," *Proceedings of the IEEE International Conference on Computer Vision*, pp. 89–96, November 2011, Barcelona, Spain.
- [114] J. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, June 1999.
- [115] W. Scheirer, A. Rocha, R. Micheals, and T. Boulton, "Robust fusion: Extreme value theory for recognition score normalization," *Proceedings of the European Conference on Computer Vision*, pp. 481–495, September 2010, Crete, Greece.
- [116] S. Liao, A. K. Jain, and S. Z. Li, "Partial face recognition: Alignment-free approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 5, pp. 1193–1205, May 2013.
- [117] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 509–522, April 2002.
- [118] A. Myronenko and X. Song, "Point set registration: Coherent point drift," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 12, pp. 2262–2275, December 2010.

- [119] M. A. Fischler and R. C. Bolles, “Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, June 1981.
- [120] E. Mortensen, H. Deng, and L. Shapiro, “A SIFT descriptor with global context,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 184–190, June 2005, San Diego, CA.
- [121] H. Moon and P. Phillips, “Computational and performance aspects of PCA-based face-recognition algorithms,” *Perception*, vol. 30, no. 3, pp. 303–321, March 2001.
- [122] Y. Yang and S. Newsam, “Geographic image retrieval using local invariant features,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 2, pp. 818–832, January 2013.
- [123] W. Tong, J. Lee, R. Jin, and A. K. Jain, “Gang and moniker identification by graffiti matching,” *Proceedings of the International ACM Workshop on Multimedia in Forensics and Intelligence*, pp. 1–6, November 2011, Scottsdale, AZ.
- [124] A. Parra, B. Zhao, J. Kim, and E. J. Delp, “Recognition, segmentation and retrieval of gang graffiti images on a mobile device,” *Proceedings of the IEEE International Conference on Technologies for Homeland Security*, pp. 178–183, November 2013, Waltham, MA.
- [125] Z. Wang and M. Liang, “Locally affine invariant descriptors for shape matching and retrieval,” *IEEE Signal Processing Letters*, vol. 17, no. 9, pp. 803–806, September 2010.
- [126] M. Daliri and V. Torre, “Robust symbolic representation for shape recognition and retrieval,” *Pattern Recognition*, vol. 41, no. 5, pp. 1782–1798, May 2008.
- [127] M. Donoser and H. Bischof, “Diffusion processes for retrieval revisited,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1320–1327, June 2013, Portland, OR.
- [128] M. Cho and K. MuLee, “Authority-shift clustering: Hierarchical clustering by authority seeking on graphs,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3193–3200, June 2010, San Francisco, CA.
- [129] B. Wang and Z. Tu, “Affinity learning via self-diffusion for image segmentation and clustering,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2312–2319, June 2012, Providence, RI.
- [130] J. Jiang, B. Wang, and Z. Tu, “Unsupervised metric learning by self-smoothing operator,” *Proceedings of the IEEE International Conference on Computer Vision*, pp. 794–801, November 2011, Barcelona, Spain.
- [131] J. Wang, Y. Li, X. Bai, Y. Zhang, C. Wang, and N. Tang, “Learning context-sensitive similarity by shortest path propagation,” *Pattern Recognition*, vol. 44, no. 1011, pp. 2367–2374, November 2011.
- [132] X. Yang, L. Prasad, and L. J. Latecki, “Affinity learning with diffusion on tensor product graph,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 28–38, January 2013.

- [133] X. Yang, S. Koknar-Tezel, and L. J. Latecki, "Locally constrained diffusion process on locally densified distance spaces with applications to shape retrieval," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 357–364, June 2009, Miami Beach, FL.
- [134] X. Bai, X. Yang, L. J. Latecki, W. Liu, and Z. Tu, "Learning context-sensitive shape similarity by graph transduction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 5, pp. 861–874, May 2010.
- [135] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: Bringing order to the web," *Technical Report*, November 1999, Stanford InfoLab.
- [136] D. Zhou, J. Weston, A. Gretton, O. Bousquet, and B. Schölkopf, "Ranking on data manifolds," *Proceedings of Neural Information Processing Systems*, pp. 169–176, December 2004, Vancouver, BC, Canada.
- [137] M. Pelillo, "Matching free trees with replicator equations," *Proceedings of Neural Information Processing Systems*, pp. 865–872, December 2002, Vancouver, BC, Canada.
- [138] X. Qian, X. Tan, Y. Zhang, R. Hong, and M. Wang, "Enhancing sketch-based image retrieval by re-ranking and relevance feedback," *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 195–208, January 2016.
- [139] J. Kim, H. Li, J. Yue, and E. J. Delp, "Shape matching using a self similar affine invariant descriptor," *Proceedings of the IEEE International Conference on Image Processing*, pp. 2470–2474, September 2016, Phoenix, AZ.
- [140] J. Kim, H. Li, Y. J., and E. J. Delp, "Tattoo image retrieval for region of interest," *Proceedings of the IEEE International Conference on Technologies for Homeland Security*, pp. 1–6, May 2016, Waltham, MA.
- [141] A. Balikai and P. M. Hall, "Depiction invariant object matching," *Proceedings of the British Machine Vision Conference*, pp. 56.1–56.11, September 2012, Surrey, UK.
- [142] N. Kohli, R. Singh, and M. Vatsa, "Self-similarity representation of weber faces for kinship classification," *Proceedings of the IEEE International Conference on Biometrics: Theory, Applications and Systems*, pp. 245–250, September 2012, Arlington, VA.
- [143] S. Kim, D. Min, B. Ham, S. Ryu, M. N. Do, and K. Sohn, "DASC: Dense adaptive self-correlation descriptor for multi-modal and multi-spectral correspondence," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2103–2112, June 2015, Boston, MA.
- [144] L. Torgo and R. Ribeiro, "Precision and recall for regression," *Proceedings of the 12th International Conference on Discovery Science, DS 09*, pp. 332–346, 2009, Porto, Portugal.
- [145] P. R. Christopher D. Manning and H. Schtze, *Introduction to Information Retrieval*. Cambridge, UK: Cambridge University Press, 2009.

- [146] J. Kim, K. Tahboub, and E. J. Delp, "Spatial pyramid alignment for sparse coding based object classification," *To appear, Proceedings of the IEEE International Conference on Image Processing*, September 2017, Beijing, China.
- [147] F. Li, R. Fergus, and P. Peronai, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories," *Computer Vision and Image Understanding*, vol. 106, no. 1, pp. 59–70, April 2007.
- [148] G. Griffin, A. Holub, and P. Perona, "Caltech-256 object category dataset," *Technical Report*, April 2007, California Institute of Technology.
- [149] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, September 2015.
- [150] G. J. J. Geusebroek, C. Veenman, and A. Smeulders, "Kernel codebooks for scene categorization," *Proceedings of the European Conference on Computer Vision*, vol. 5304, pp. 696–709, October 2008, Marseille, France.
- [151] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3360–3367, June 2010, San Francisco, CA.
- [152] Y. Jia, C. Huang, and T. Darrell, "Beyond spatial pyramids: Receptive field learning for pooled image features," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3370–3377, June 2012, Providence, RI.
- [153] E. Boser, M. Isabelle, and V. Vapnik, "A training algorithm for optimal margin classifiers," *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, pp. 144–152, July 1992, Pittsburgh, PA.
- [154] C. C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, September 1995.
- [155] H. Lee, A. Battle, R. Raina, and A. Ng, "Efficient sparse coding algorithms," *Proceedings of Neural Information Processing Systems*, pp. 801–808, December 2006, Vancouver, BC, Canada.
- [156] Y. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning mid-level features for recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2559–2566, June 2010, San Francisco, CA.
- [157] C. Liu, J. Yuen, and A. Torralba, "SIFT flow: Dense correspondence across scenes and its applications," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 978–994, May 2011.
- [158] J. Kim, A. Parra, J. Yue, H. Li, and E. J. Delp, "Robust local and global shape context for tattoo image matching," *Proceedings of the IEEE International Conference on Image Processing*, pp. 2194–2198, September 2015, Quebec city, Quebec, Canada.

- [159] Z. Jiang, Z. Lin, and L. Davis, "Learning a discriminative dictionary for sparse coding via label consistent K-SVD," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1697–1704, June 2011, Colorado Springs, CO.
- [160] H. Fan, Y. Cong, and Y. Tang, "Self-closed partial shape descriptor for shape retrieval," *Proceedings of the IEEE International Conference on Image Processing*, pp. 505–508, September 2012, Orlando, FL.
- [161] J. Wang, X. Bai, X. You, W. Liu, and L. Latecki, "Shape matching and classification using height functions," *Pattern Recognition Letters*, vol. 33, no. 2, pp. 134–143, January 2012.
- [162] F. Mokhtarian, S. Abbasi, and J. Kittler, "Efficient and robust retrieval by shape content through curvature scale space," *Image Databases and Multimedia Search*. River Edge, NJ: World Scientific Publishing Co., Inc., 1998, pp. 51–58.
- [163] Y. Liu, Q. Jia, H. Guo, and X. Fan, "A shape matching framework using metric partition constraint," *Proceedings of the IEEE International Conference on Image Processing*, pp. 3494–3498, September 2013, Melbourne, Australia.
- [164] H. Ling and D. Jacobs, "Shape classification using the inner distance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 2, pp. 286–299, February 2007.
- [165] M. McNeill and S. Vijayakumar, "Hierarchical procrustes matching for shape retrieval," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 885–894, June 2006, New York, NY.
- [166] P. Felzenszwalb and J. Schwartz, "Hierarchical matching of deformable shapes," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2007, Minneapolis, MN.
- [167] C. Xu, J. Liu, and X. Tang, "2D shape matching by contour flexibility," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 1, pp. 180–186, January 2009.
- [168] N. Alajlan, M. Kamel, and G. Freeman, "Geometry-based image retrieval in binary image databases," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 6, pp. 1003–1013, June 2008.
- [169] Z. Tu and A. Yuille, "Shape matching and recognition - using generative models and informative features," *Proceedings of the European Conference on Computer Vision*, pp. 195–209, May 2004, prague, Czech Republic. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-24672-5_16
- [170] H. Chui and A. Rangarajan, "A new point matching algorithm for non-rigid registration," *Computer Vision and Image Understanding*, vol. 89, no. 2-3, pp. 114–141, February 2003. [Online]. Available: [http://dx.doi.org/10.1016/S1077-3142\(03\)00009-2](http://dx.doi.org/10.1016/S1077-3142(03)00009-2)
- [171] L. Chai, Z. Qin, and Q. Li, "Ordered histogram of shapemes: An ordered bag-of-features based shape descriptor for efficient shape matching," *Proceedings of the IEEE International Conference on Image Processing*, pp. 2929–2933, September 2013, Melbourne, Australia.

- [172] Y. Cao, C. Wang, Z. Li, L. Zhang, and L. Zhang, "Spatial-bag-of-features," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3352–3359, June 2010, San Francisco, CA.
- [173] R. Penrose, "A generalized inverse for matrices," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 51, no. 3, pp. 406–413, July 1955.
- [174] X. Yang, X. Bai, L. J. Latecki, and Z. Tu, "Improving shape retrieval by learning graph transduction," *Proceedings of the European Conference on Computer Vision*, pp. 788–801, October 2008, Marseille, France.
- [175] L. J. Latecki, R. Lakamper, and U. Eckhardt, "Shape descriptors for non-rigid shapes with a single closed contour," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 424–429, June 2000, Hilton Head Island, SC.
- [176] A. Parra, M. Boutin, and E. J. Delp, "Location-aware gang graffiti acquisition and browsing on a mobile device," *Proceedings of the IS&T/SPIE Electronic Imaging on Multimedia on Mobile Devices*, pp. 830 402–1–13, January 2012, San Francisco, CA.
- [177] A. Parra, "Integrated mobile systems using image analysis with applications in public safety," Ph.D. dissertation, Purdue University, West Lafayette, IN, August 2014.

VITA

VITA

Joonsoo Kim was born in Seoul, Korea. He received the B.S. in Electrical and Electronics Engineering (EE) from Yonsei University, Seoul, Korea. He also received M.S. in Electrical and Electronics Engineering (EE) from Yonsei University, Seoul, Korea. Mr. Kim joined the Ph.D. program at the School of Electrical and Computer Engineering, Purdue University, West Lafayette, Indiana in August 2012. He has worked as Research Assistant in the Video and Image Processing Laboratory (VIPER) under the supervision of Professor Edward J. Delp since 2013. He was an intern at the advanced technology group (ATG) of Dolby Laboratory, Los Angeles, CA and at the camera system algorithm group of Qualcomm Inc., San Diego, CA in the summers of 2015 and 2016 respectively. His research interests are image processing, computer vision, machine learning, and video compression. He is a student member of the IEEE and the IEEE Signal Processing Society.