Computer Vision for Embedded Systems

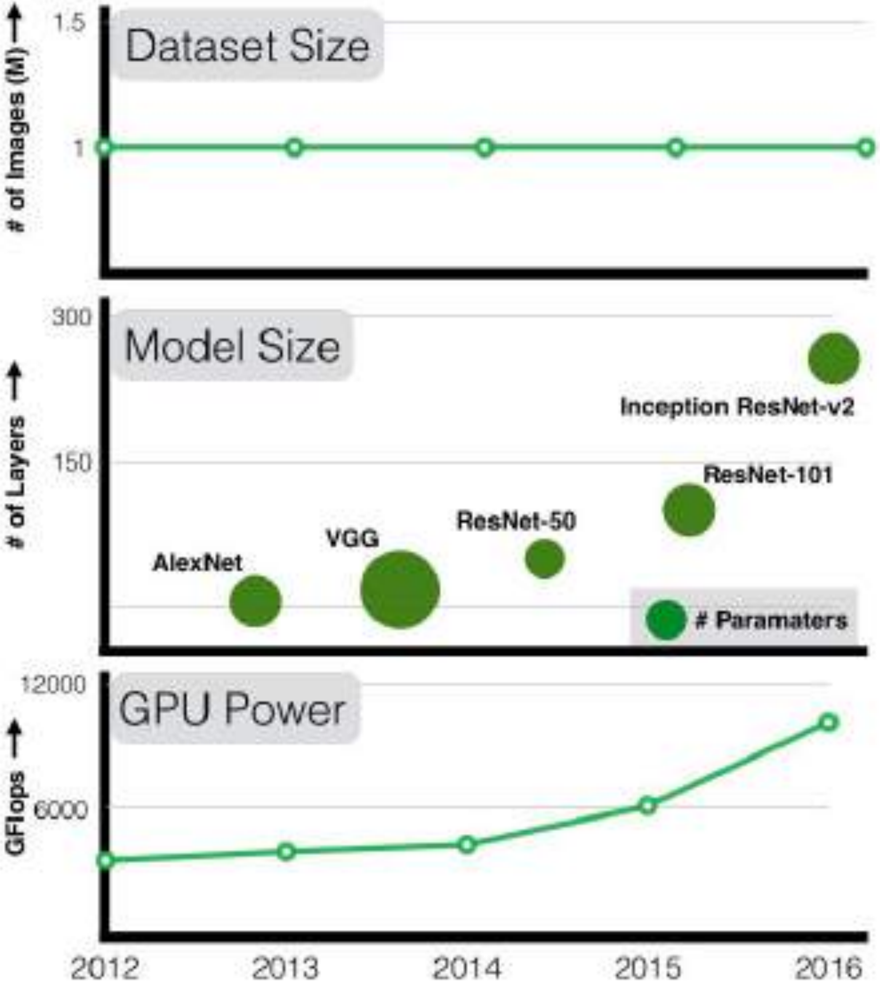
Yung-Hsiang Lu
Purdue University
yunglu@purdue.edu

PURDUE
UNIVERSITY.

Revisiting Unreasonable Effectiveness of Data in Deep Learning Era

Chen Sun, Abhinav Shrivastava, Saurabh Singh, Abhinav Gupta, ICCV 2017

JFT dataset: 300M images, 18,291 categories



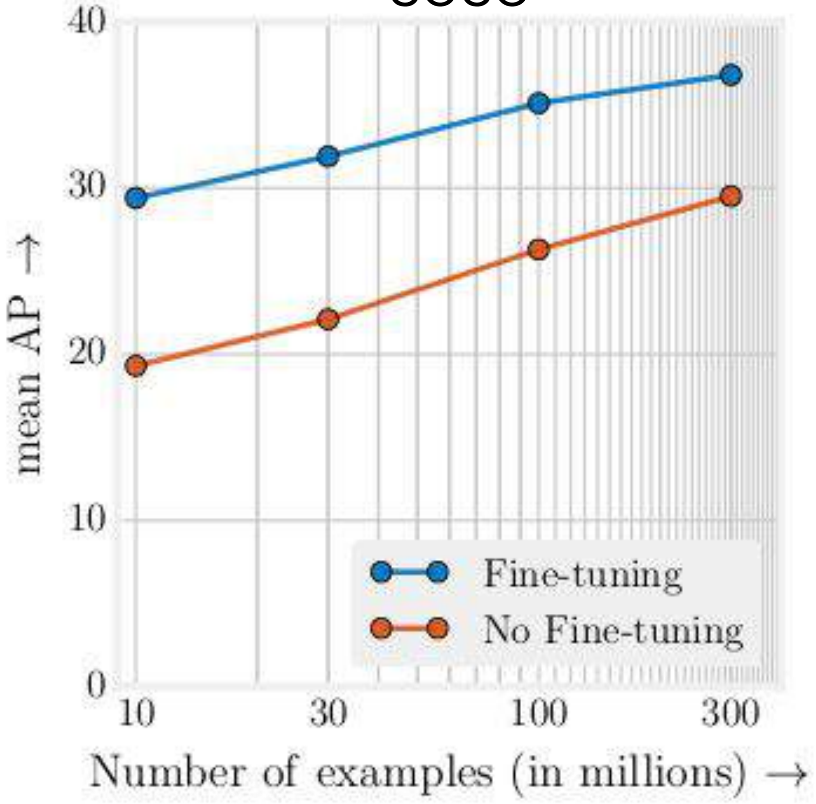
Dataset Size
# of Images (M)
1.5
1
Model Size
# of Layers
300
150
AlexNet
VGG
ResNet-50
ResNet-101
Inception ResNet-v2
Paramaters
GPU Power
GFlops
12000
6000
2012
2013
2014
2015
2016

JFT Dataset

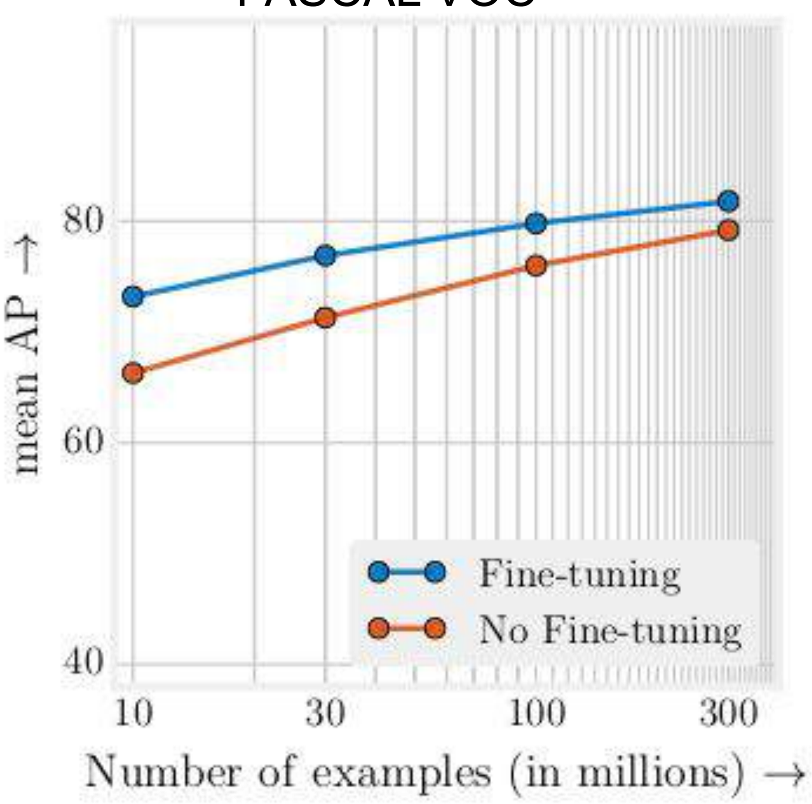
- 300M images
- 375M labels
- 18,291 categories
  - 1,165 types of animals
  - 5,720 types of vehicles
  - maximum depth of hierarchy is 12
  - maximum number of children is 2,876
- heavy tail distribution: 3K categories with fewer than 100 images each
- image sizes: 340 x 340 cropped to 299 x 299, normalized to [-1, 1]

Effects of Training Examples

COCO



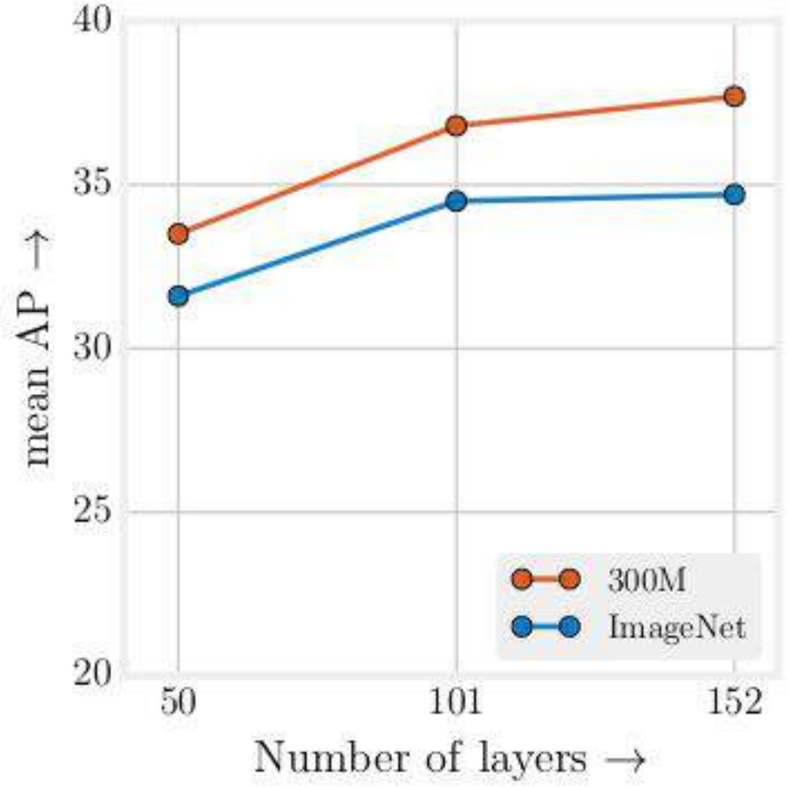
PASCAL VOC



Effect of Model Capacity

COCO
on ResNet

#Layers	ImageNet	300M
50	31.6	33.5
101	34.5	36.8
152	34.7	37.7

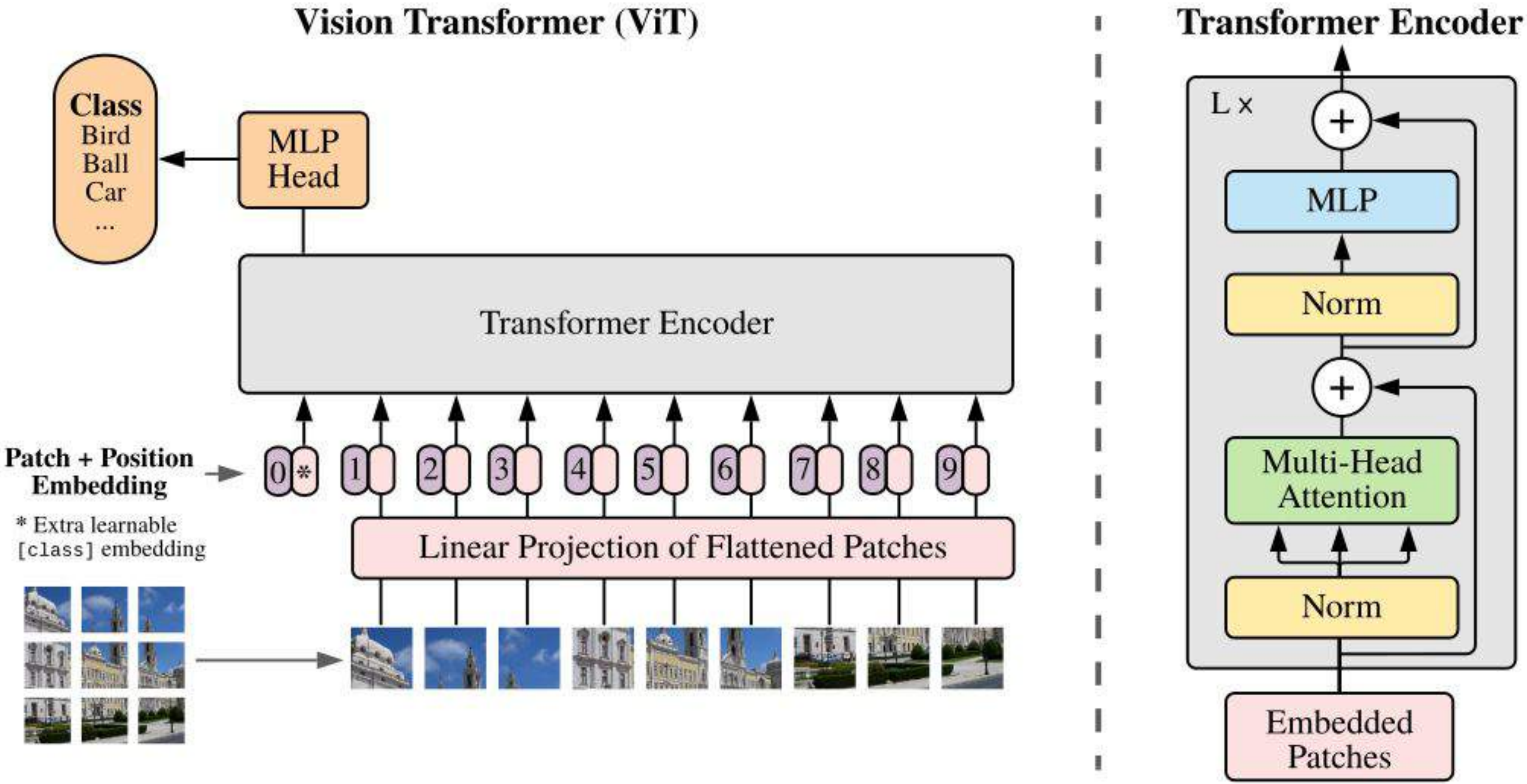


Limitations of Convolutional Neural Networks

- Convolution considers neighbor pixels but at fixed distances
- Same parameters are applied to all pixels even though objects may have different sizes
- Hyperparameters (stride, filter size, number of layers …) determined in advance (may be determined by neural architecture search)

An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby 2020


Vision Transformer (ViT)
Class
Bird
Ball
Car
...
MLP Head
Transformer Encoder
Patch + Position Embedding
* Extra learnable [class] embedding
0 *
1
2
3
4
5
6
7
8
9
Linear Projection of Flattened Patches
Transformer Encoder
L ×
+
MLP
Norm
+
Multi-Head Attention
Norm
Embedded Patches

Create Image Patches

$$R^{N \times W \times C} \Rightarrow R^{N \times (P^2 \times C)}$$

H image height

W image width

C number of channels

P size of patch

$$N = \frac{HW}{P^2}$$ number of patches

Options of Position Embedding

- no position information (bags of patches)
- 1D position embedding (sequences of patches)
- 2D position embedding
- relative position embedding

Pos. Emb.	Default/Stem	Every Layer	Every Layer-Shared
No Pos. Emb.	0.61382	N/A	N/A
1-D Pos. Emb.	0.64206	0.63964	0.64292
2-D Pos. Emb.	0.64001	0.64046	0.64022
Rel. Pos. Emb.	0.64032	N/A	N/A

Datasets

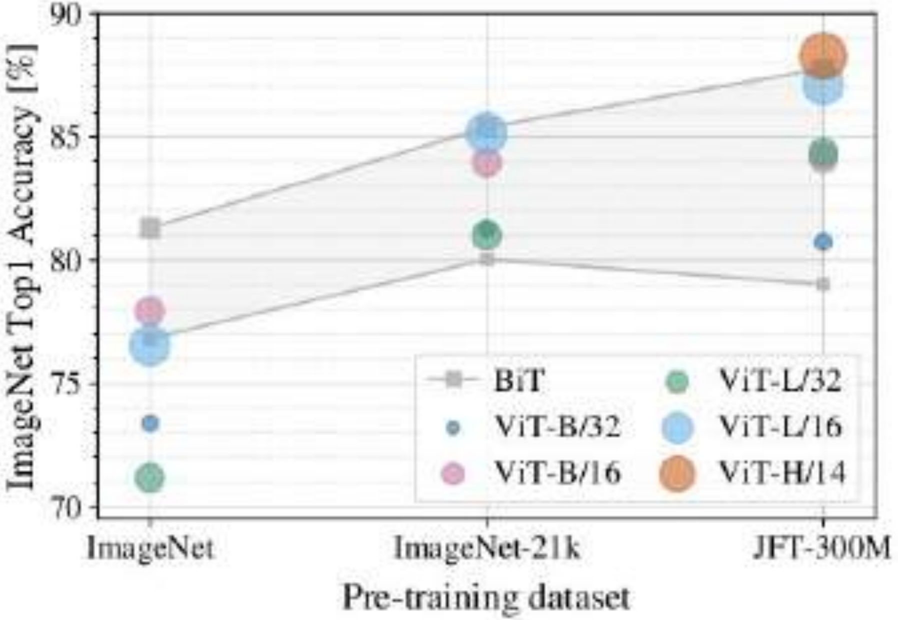
- ImageNet 1K: 1K classes, 1.3M images
- ImageNet 21K: 21K classes, 14M images
- JFT: 18K classes, 14M images
- CIFAR-10 and 100
- Oxford-IIIT Pets
- Oxford Flowers-102

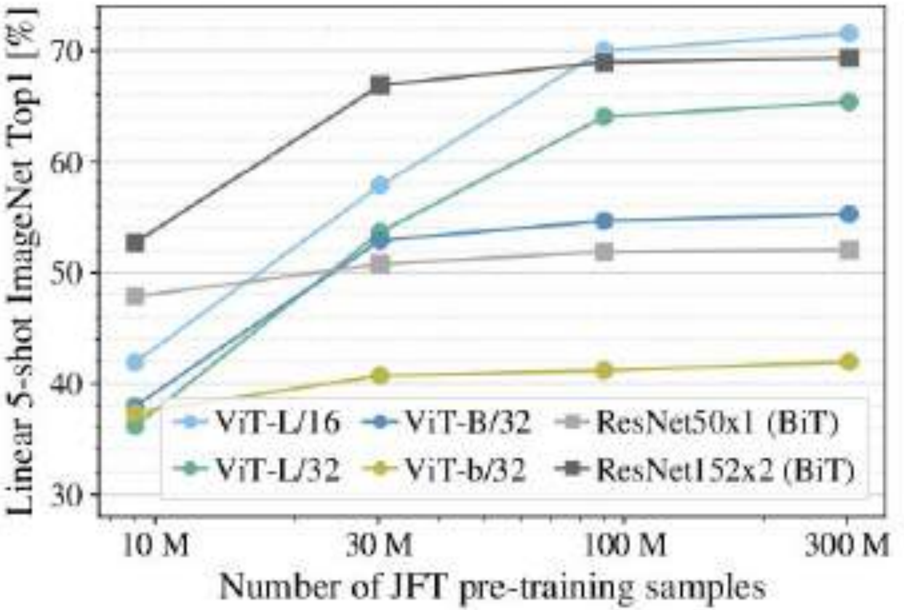
Model Variants

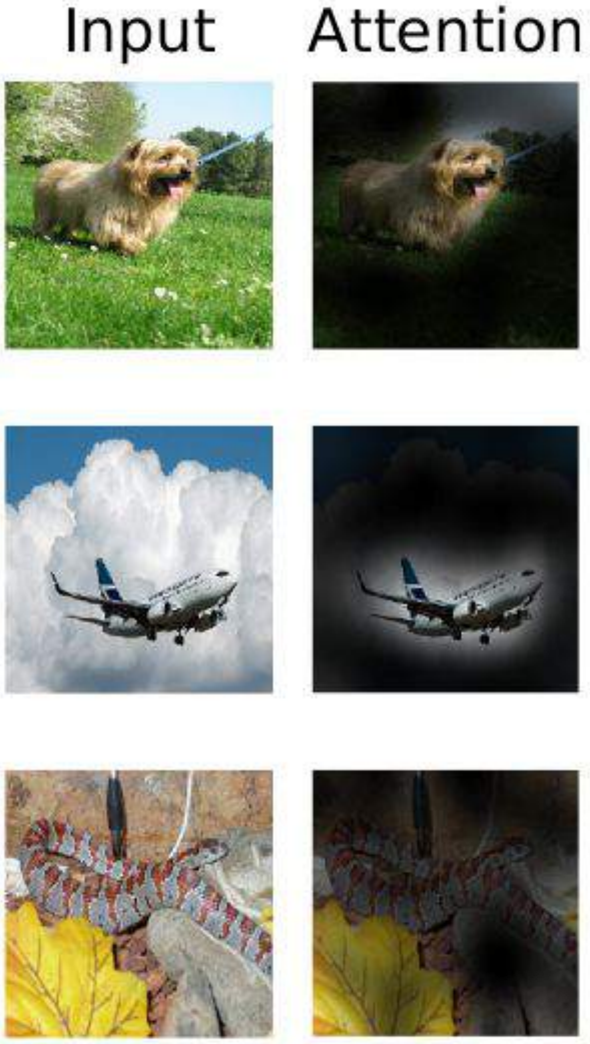
Model	Layers	Hidden size D	MLP size	Heads	Params
ViT-Base	12	768	3072	12	86M
ViT-Large	24	1024	4096	16	307M
ViT-Huge	32	1280	5120	16	632M

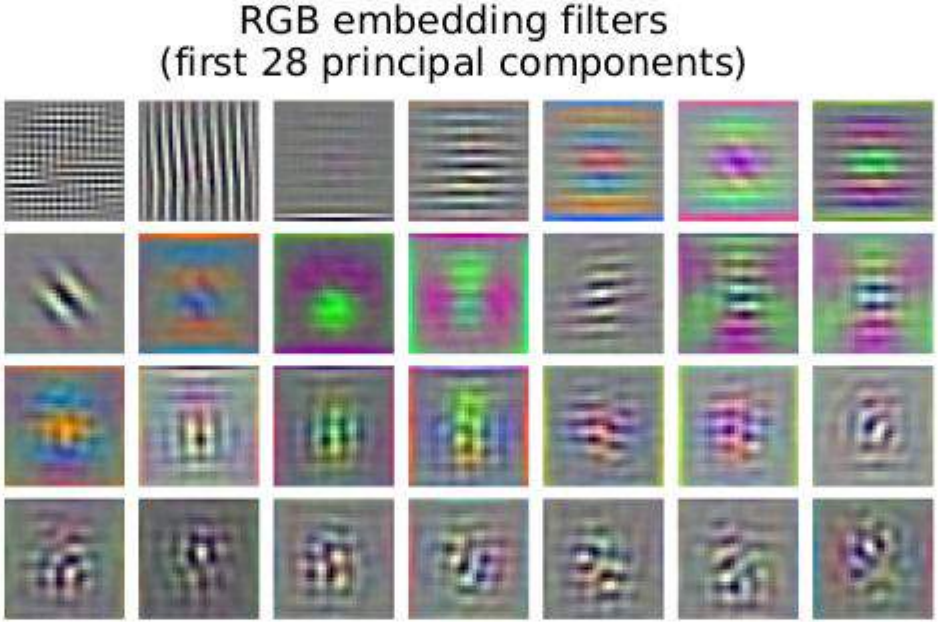
Comparison

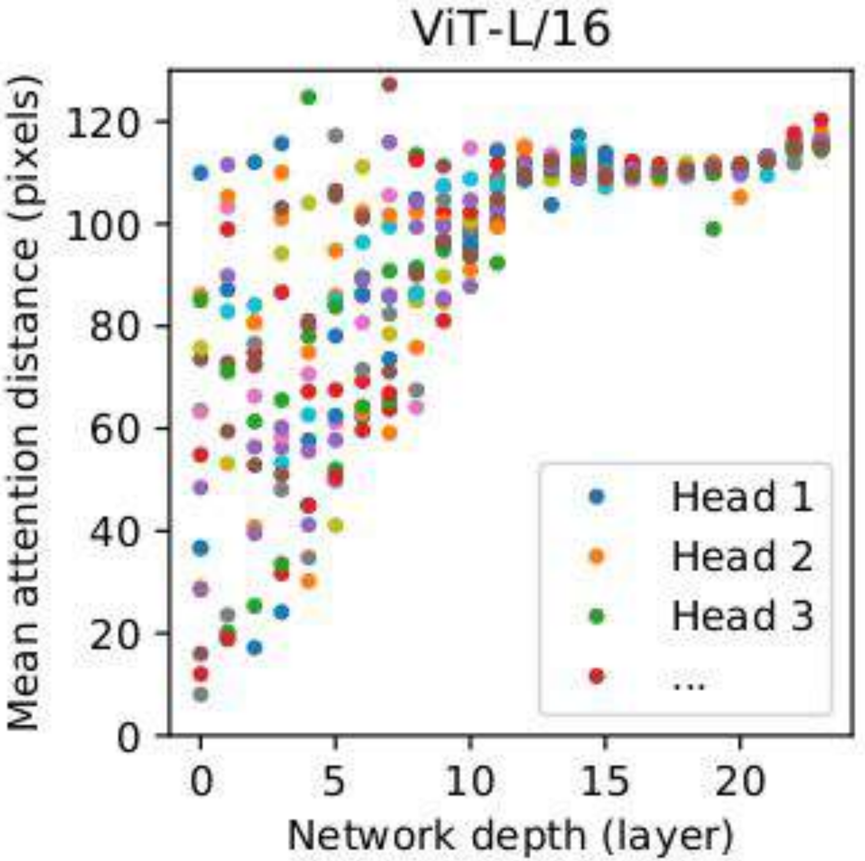
	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k


ImageNet Top1 Accuracy [%]
90
85
80
75
70
ImageNet
ImageNet-21k
JFT-300M
Pre-training dataset
BiT
ViT-B/32
ViT-B/16
ViT-L/32
ViT-L/16
ViT-H/14


Linear 5-shot ImageNet Top1 [%]
70
60
50
40
30
10 M
30 M
100 M
300 M
Number of JFT pre-training samples
ViT-L/16
ViT-B/32
ResNet50x1 (BiT)
ViT-L/32
ViT-b/32
ResNet152x2 (BiT)

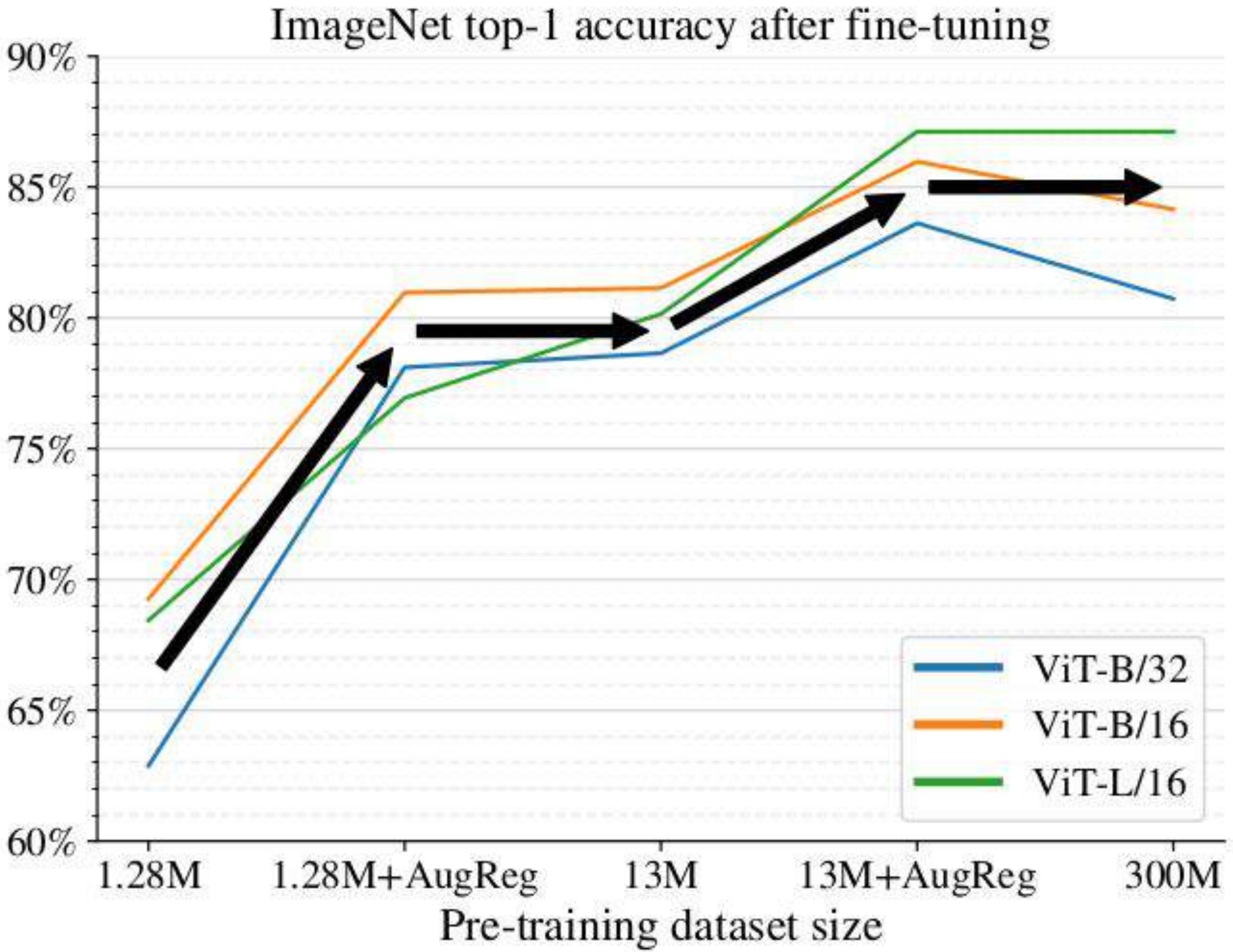

Input
Attention


RGB embedding filters
(first 28 principal components)


ViT-L/16
Mean attention distance (pixels)
Network depth (layer)
0
20
40
60
80
100
120
0
5
10
15
20
Head 1
Head 2
Head 3
...

How to train your ViT? Data, Augmentation, and Regularization in Vision Transformers

Andreas Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, Lucas Beyer 2021


ImageNet top-1 accuracy after fine-tuning
90%
85%
80%
75%
70%
65%
60%
1.28M
1.28M+AugReg
13M
13M+AugReg
300M
Pre-training dataset size
ViT-B/32
ViT-B/16
ViT-L/16

ViViT: A Video Vision Transformer

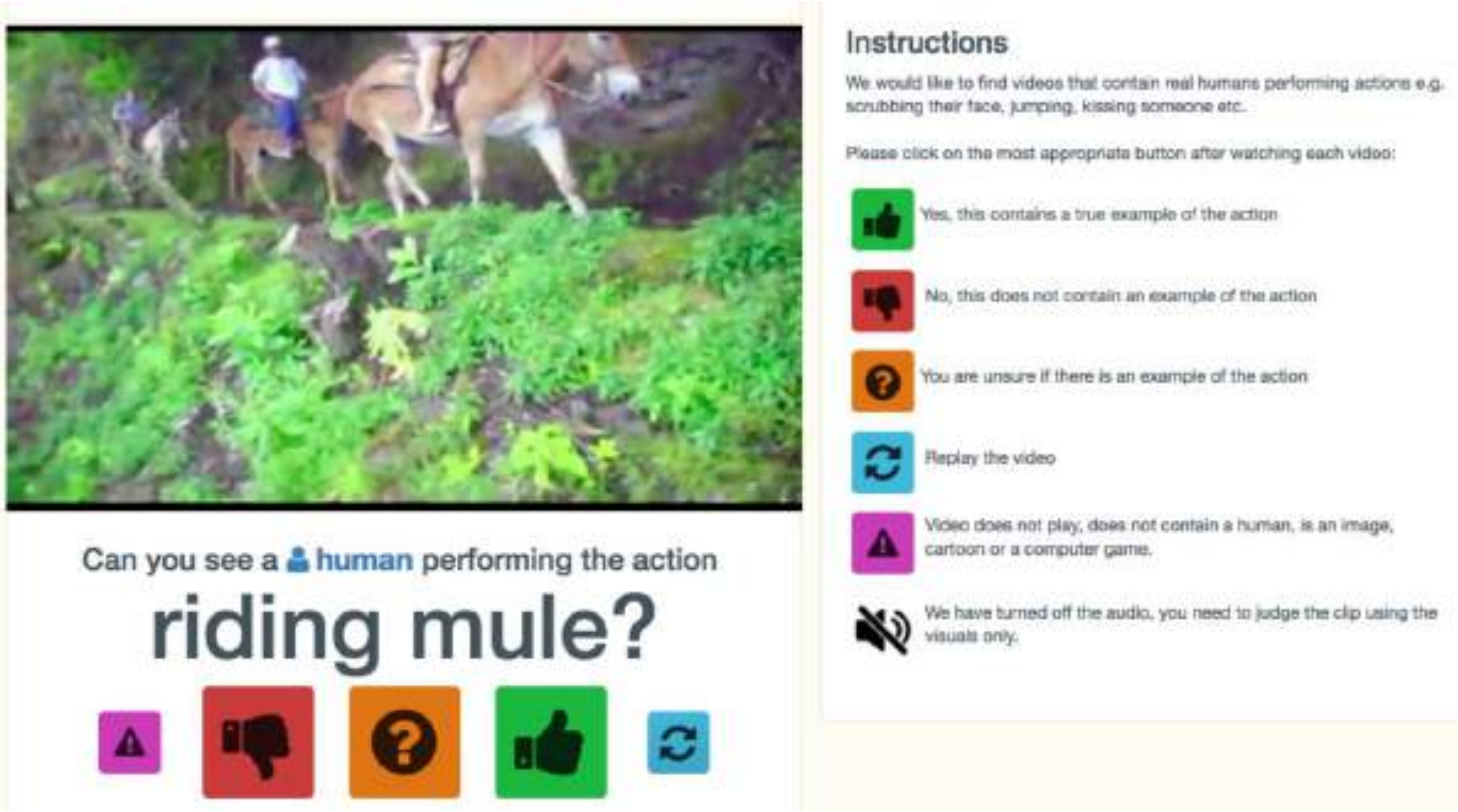
Anurag Arnab, Mostafa Dehghani, Georg Heigold Chen Sun, CVPR 2021

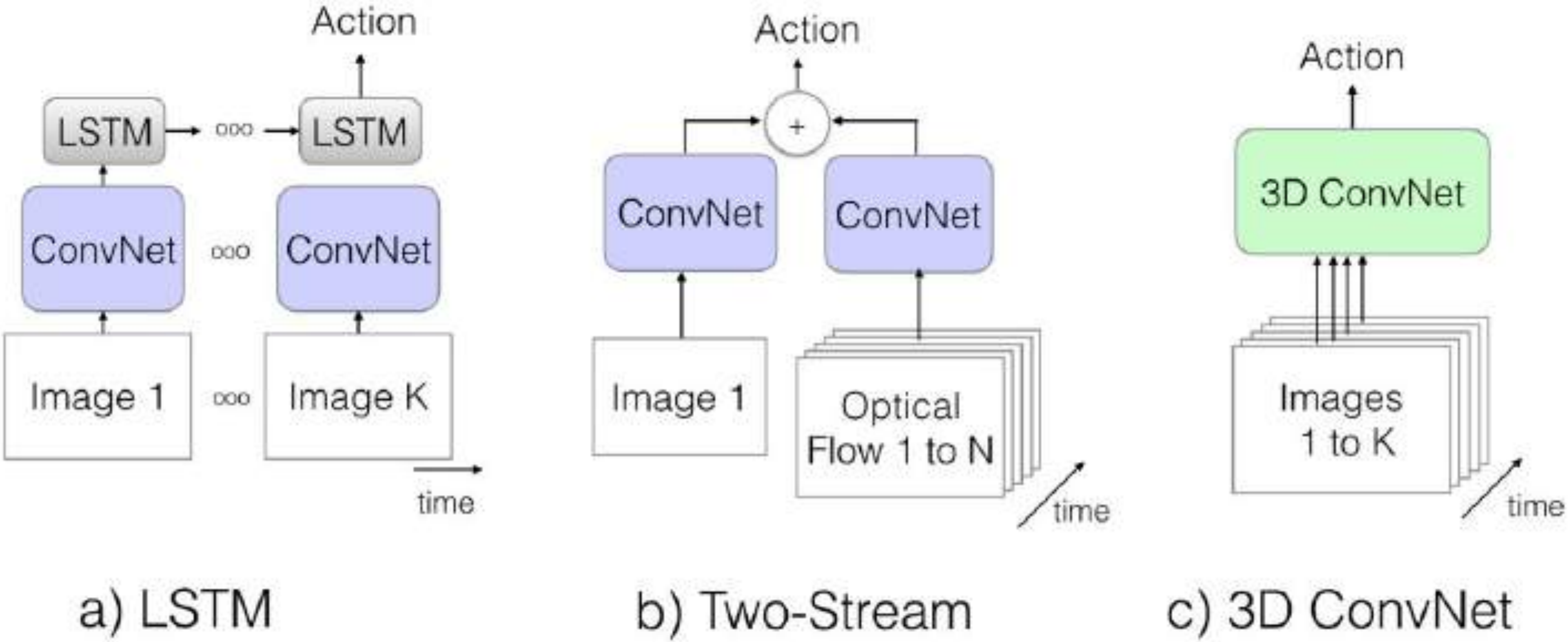
The Kinetics Human Action Video Dataset 2017

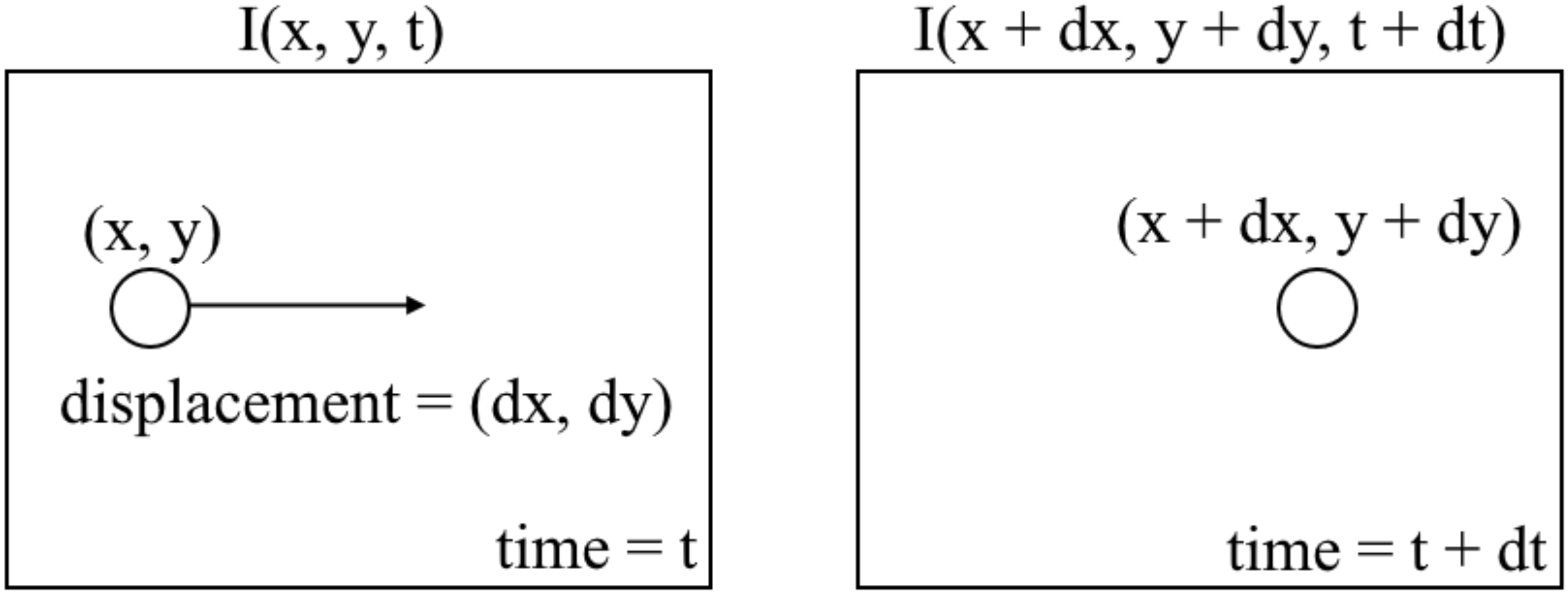
Kinetics Dataset

- 400 human action classes
- each action at least 400 clips
- each clip 10 seconds from Youtube
- Single person activities: drawing, laughing, drinking
- Person-person activities: shaking hands, hugging
- Person-object activities: washing dishes, mowing lawn

Crowdsourcing to label data

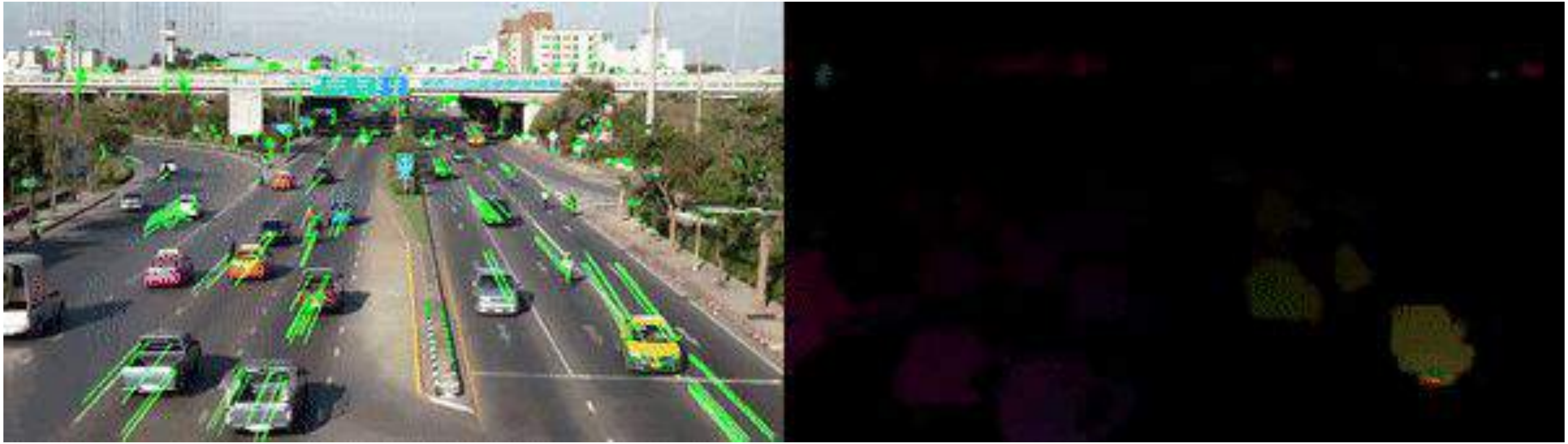



Action
LSTM
ooo
LSTM
ConvNet
ooo
ConvNet
Image 1
ooo
Image K
time
Action
+
ConvNet
ConvNet
Image 1
Optical Flow 1 to N
time
Action
3D ConvNet
Images 1 to K
time
a) LSTM
b) Two-Stream
c) 3D ConvNet



$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t)$$

https://nanonets.com/blog/optical-flow/



https://nanonets.com/blog/optical-flow/

$$I(x+\delta x, y+\delta y, t+\delta t) = I(x,y,t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t + \ldots$$

$$\Rightarrow \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = 0$$

$$\frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t} = 0$$

$$u = \frac{dx}{dt} \qquad v = \frac{dy}{dt}$$

https://nanonets.com/blog/optical-flow/

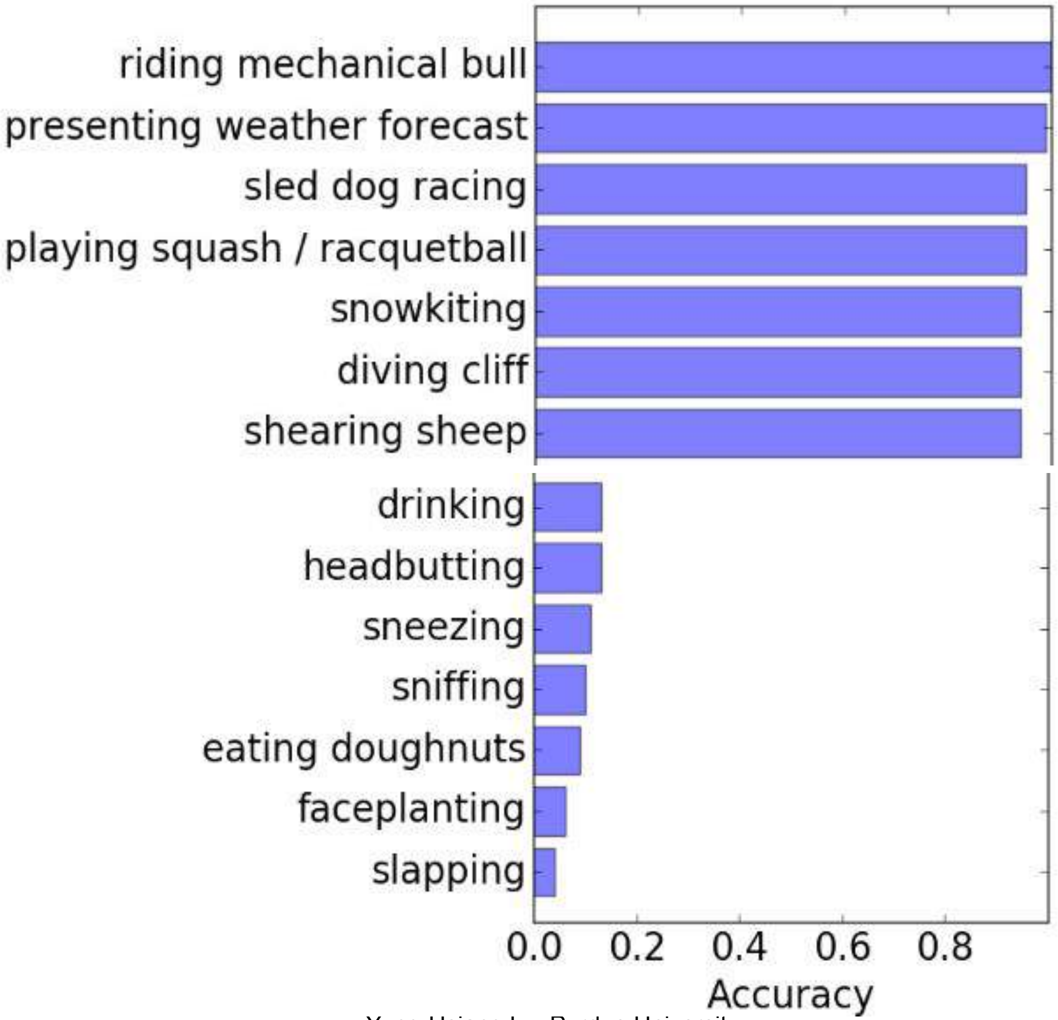
Lucas–Kanade method

The method assume the optical flow in each small neighborhood is unchanged.

$$
\begin{aligned}
I_x(q_1)V_x + I_y(q_1)V_y &= -I_t(q_1) \\
I_x(q_2)V_x + I_y(q_2)V_y &= -I_t(q_2) \\
&\vdots \\
I_x(q_n)V_x + I_y(q_n)V_y &= -I_t(q_n)
\end{aligned}
$$

$$V_x = u(\text{in the previous slide})$$

https://en.wikipedia.org/wiki/Lucas%E2%80%93Kanade_method


riding mechanical bull
presenting weather forecast
sled dog racing
playing squash / racquetball
snowkiting
diving cliff
shearing sheep
drinking
headbutting
sneezing
sniffing
eating doughnuts
faceplanting
slapping
0.0
0.2
0.4
0.6
0.8
Accuracy

Class 1	Class 2	confusion
'riding mule'	'riding or walking with horse'	40%
'hockey stop'	'ice skating'	36%
'swing dancing'	'salsa dancing'	36%
'strumming guitar'	'playing guitar'	35%
'shooting basketball'	'playing basketball'	32%
'cooking sausages'	'cooking chicken'	29%
'sweeping floor'	'mopping floor'	27%
'triple jump'	'long jump'	26%
'doing aerobics'	'zumba'	26%
'petting animal (not cat)'	'feeding goats'	25%
'shaving legs'	'waxing legs'	25%
'snowboarding'	'skiing (not slalom or crosscountry)'	22%

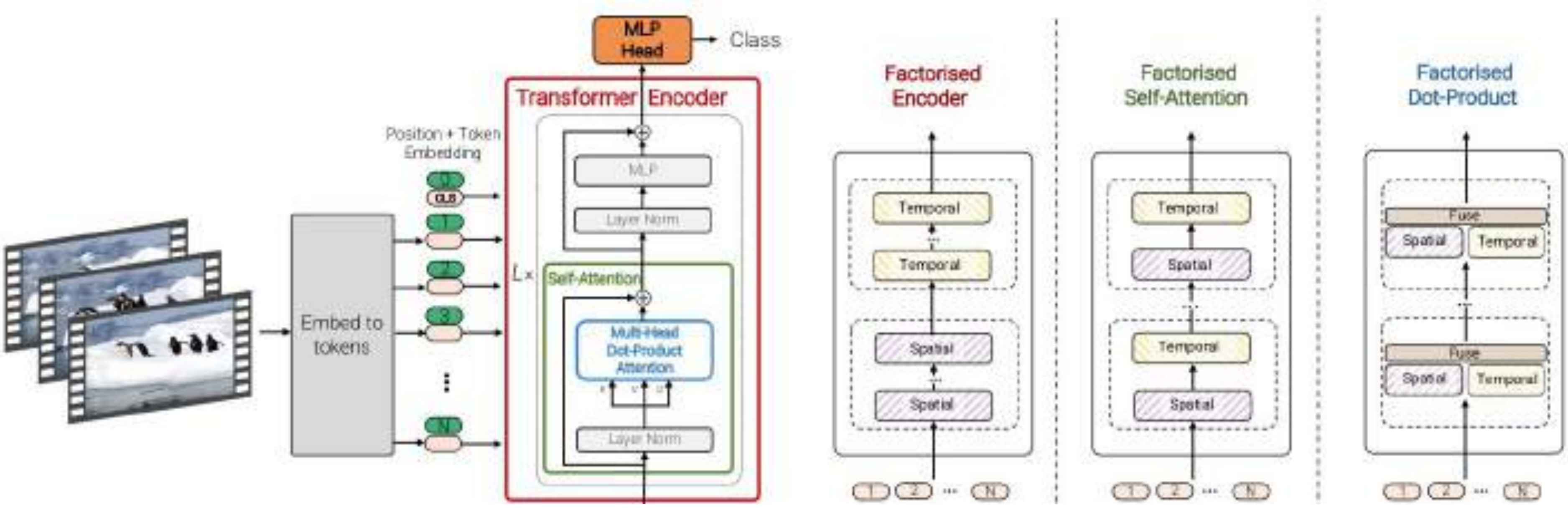
Architecture	UCF-101			HMDB-51			Kinetics		
	RGB	Flow	RGB+Flow	RGB	Flow	RGB+Flow	RGB	Flow	RGB+Flow
(a) ConvNet+LSTM	84.3	–	–	43.9	–	–	57.0 / 79.0	–	–
(b) Two-Stream	84.2	85.9	92.5	51.0	56.9	63.7	56.0 / 77.3	49.5 / 71.9	61.0 / 81.3
(c) 3D-ConvNet	51.6	–	–	24.3	–	–	56.1 / 79.5	–	–

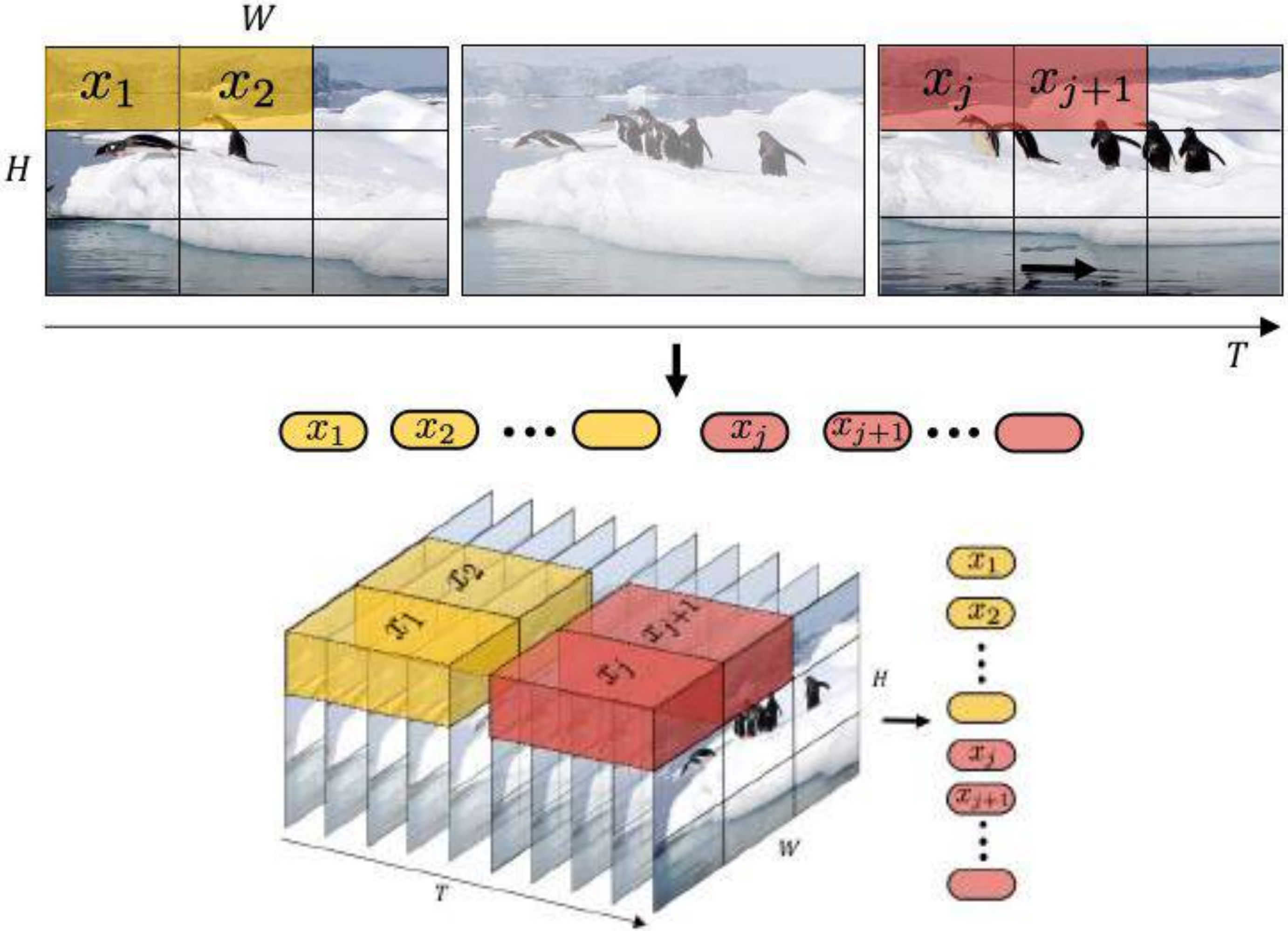
UCF-101

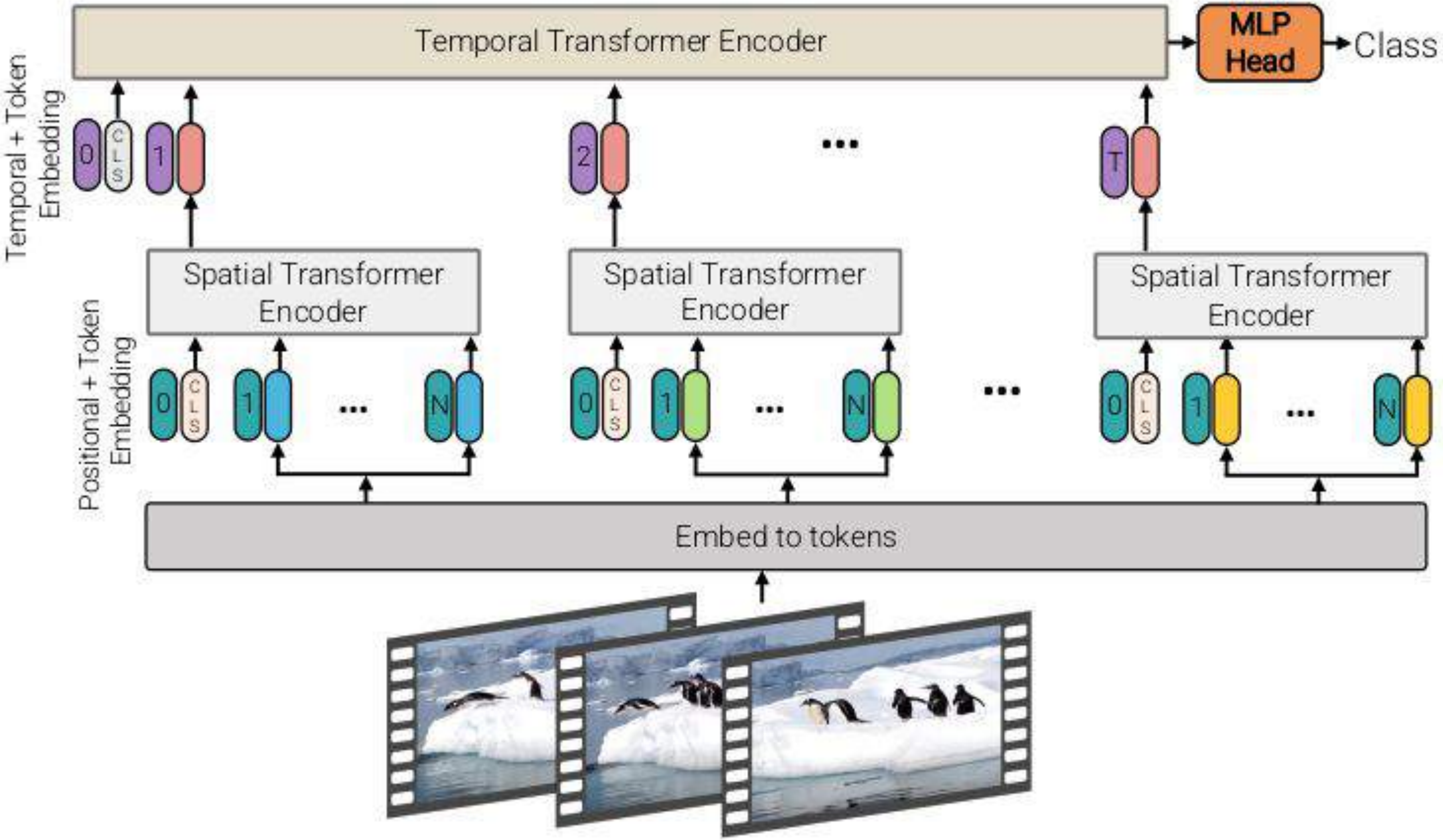
HMDB-51

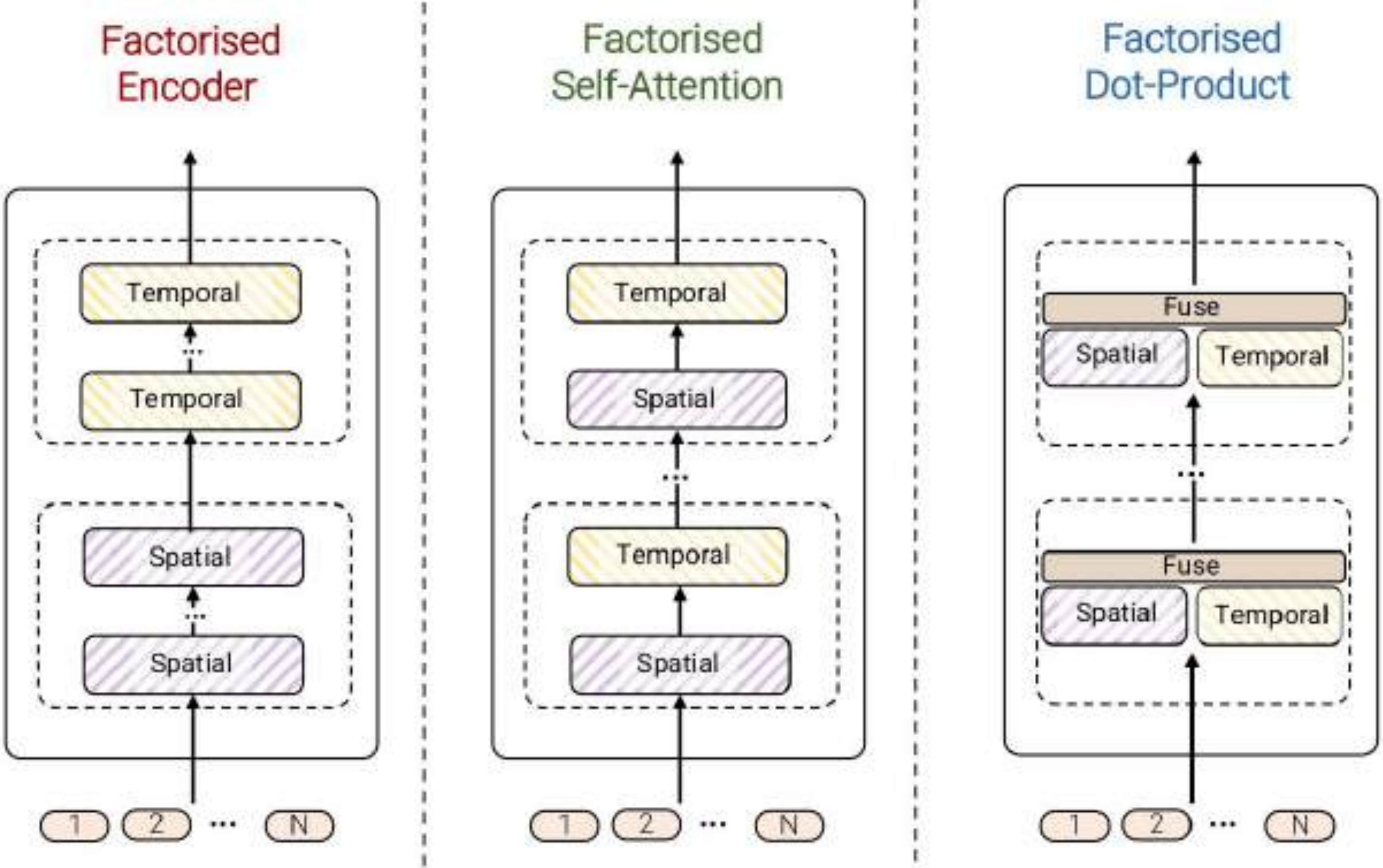
ViViT: A Video Vision Transformer

Anurag Arnab, Mostafa Dehghani, Georg Heigold Chen Sun, CVPR 2021


MLP Head
Class
Transformer Encoder
Position + Token Embedding
CLS
Embed to tokens
MLP
Layer Norm
L ×
Self-Attention
Multi-Head Dot-Product Attention
Layer Norm
Factorised Encoder
Temporal
Temporal
Spatial
Spatial
Factorised Self-Attention
Temporal
Spatial
Temporal
Spatial
Factorised Dot-Product
Fuse
Spatial
Temporal
Fuse
Spatial
Temporal


W
H
x_1
x_2
x_j
x_{j+1}
T
x_1
x_2
x_j
x_{j+1}
x_1
x_2
x_j
x_{j+1}
H
W
T
x_1
x_2
x_j
x_{j+1}


Temporal Transformer Encoder
MLP Head
Class
Temporal + Token Embedding
0
CLS
1
2
...
T
Spatial Transformer Encoder
Spatial Transformer Encoder
Spatial Transformer Encoder
Positional + Token Embedding
0
CLS
1
...
N
Embed to tokens


Factorised
Encoder
Temporal
Temporal
Spatial
Spatial
1
2
N
Factorised
Self-Attention
Temporal
Spatial
Temporal
Spatial
1
2
N
Factorised
Dot-Product
Fuse
Spatial
Temporal
Fuse
Spatial
Temporal
1
2
N

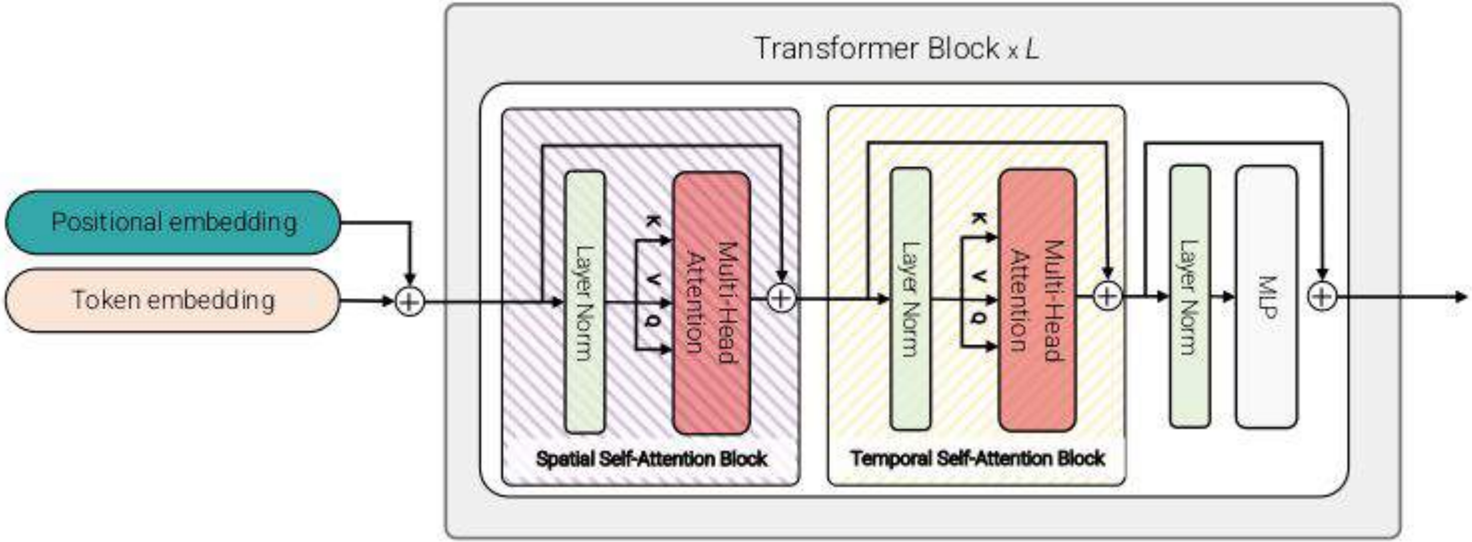
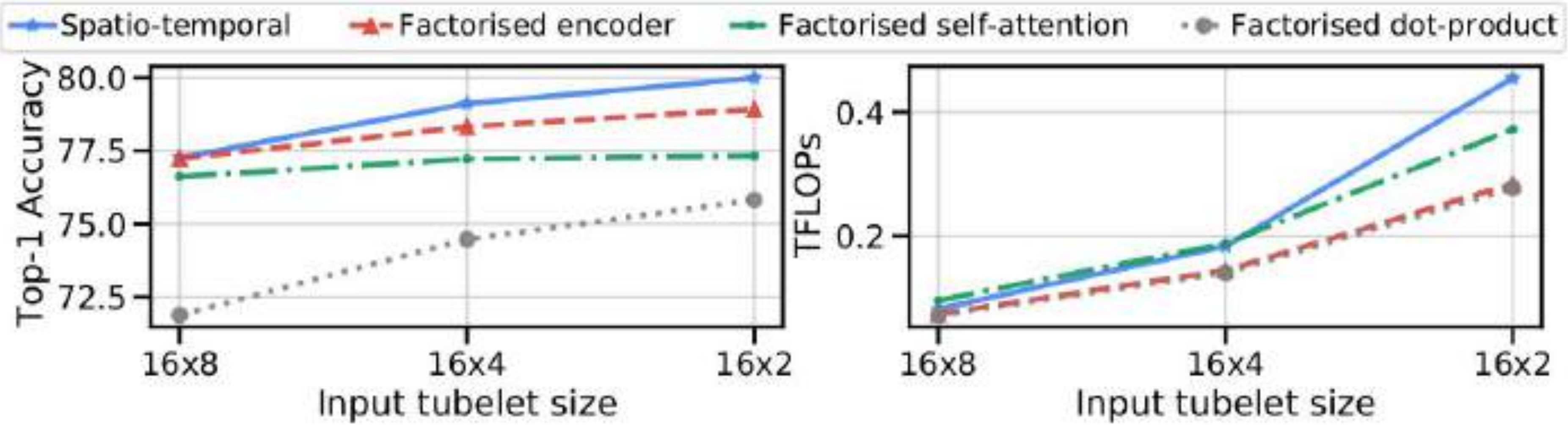

Transformer Block x L
Positional embedding
Token embedding
Layer Norm
k
v
q
Multi-Head Attention
Spatial Self-Attention Block
Layer Norm
k
v
q
Multi-Head Attention
Temporal Self-Attention Block
Layer Norm
MLP

Table 2: Comparison of model architectures using ViViT-B as the backbone, and tubelet size of 16×2. We report Top-1 accuracy on Kinetics 400 (K400) and action accuracy on Epic Kitchens (EK). Runtime is during inference on a TPU-v3.

	K400	EK	FLOPs ($\times 10^9$)	Params ($\times 10^6$)	Runtime (ms)
Model 1: Spatio-temporal	80.0	43.1	455.2	88.9	58.9
Model 2: Fact. encoder	78.8	43.7	284.4	115.1	17.4
Model 3: Fact. self-attention	77.4	39.1	372.3	117.3	31.7
Model 4: Fact. dot product	76.3	39.5	277.1	88.9	22.9
Model 2: Ave. pool baseline	75.8	38.8	283.9	86.7	17.3



(a) Accuracy (b) Compute