Soutenance de Stage de L3

Multimodal Learning: A case study for Gesture and Audio-Visual Speech Recognition

Hsieh Yu-Guan (Info 2016)

Supervised by
Amélie Cordier & Mathieu Lefort

Internship period: 14^{th} June 2017 – 11^{th} August 2017

behaviors.ai

Introduction

- BEHAVIORS.AI
- Hoomano & LIRIS
- TensorFlow

Introduction

- Artifical Intelligence

Introduction

- Artifical Intelligence
- $>$ Robotics (embodied paradigme)

Introduction

- Artifical Intelligence
- $>$ Robotics (embodied paradigme)
- $>$ Developmental robotics

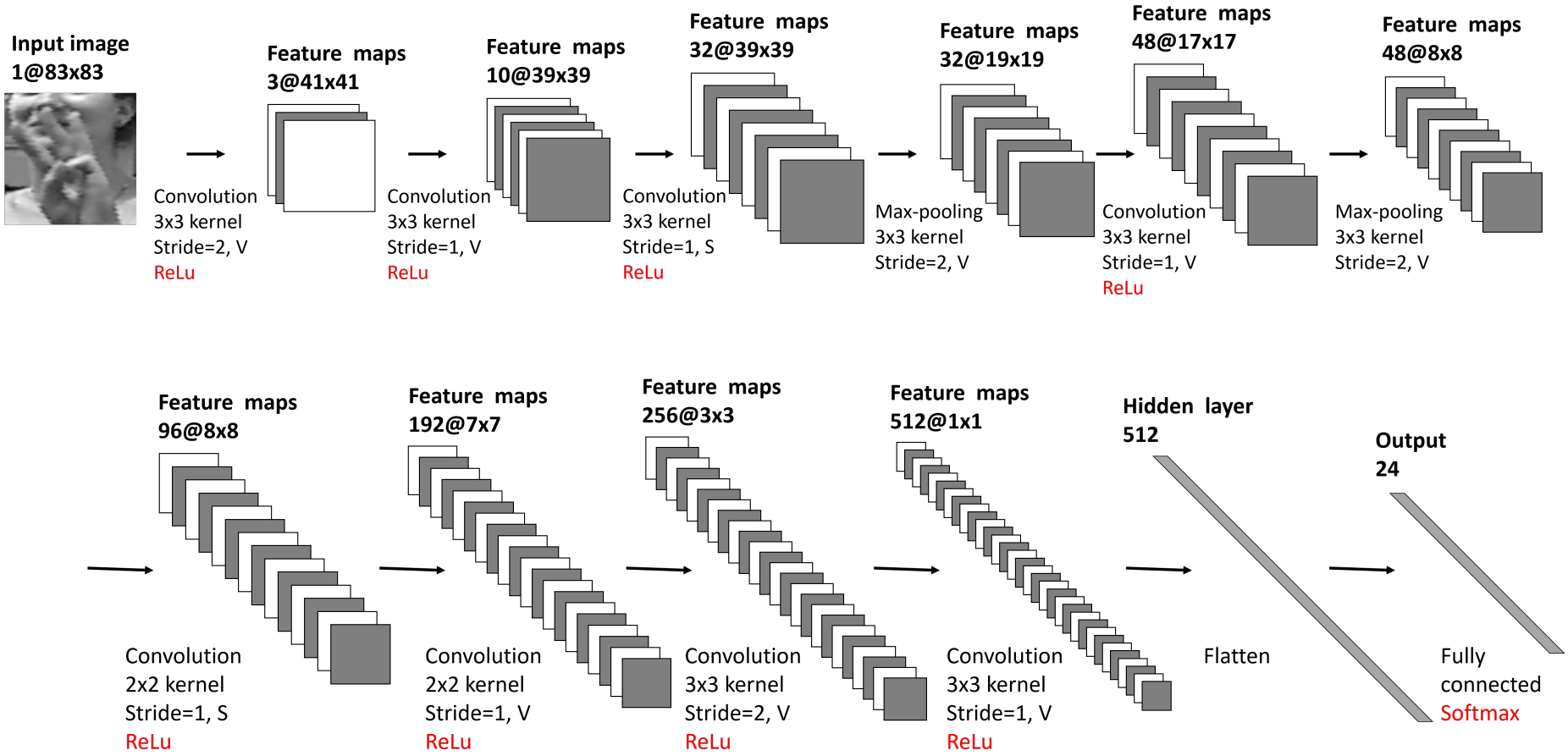
Introduction

- Artifical Intelligence
- $>$ Robotics (embodied paradigme)
- $>$ Developmental robotics
- $>$ Multimodal learning

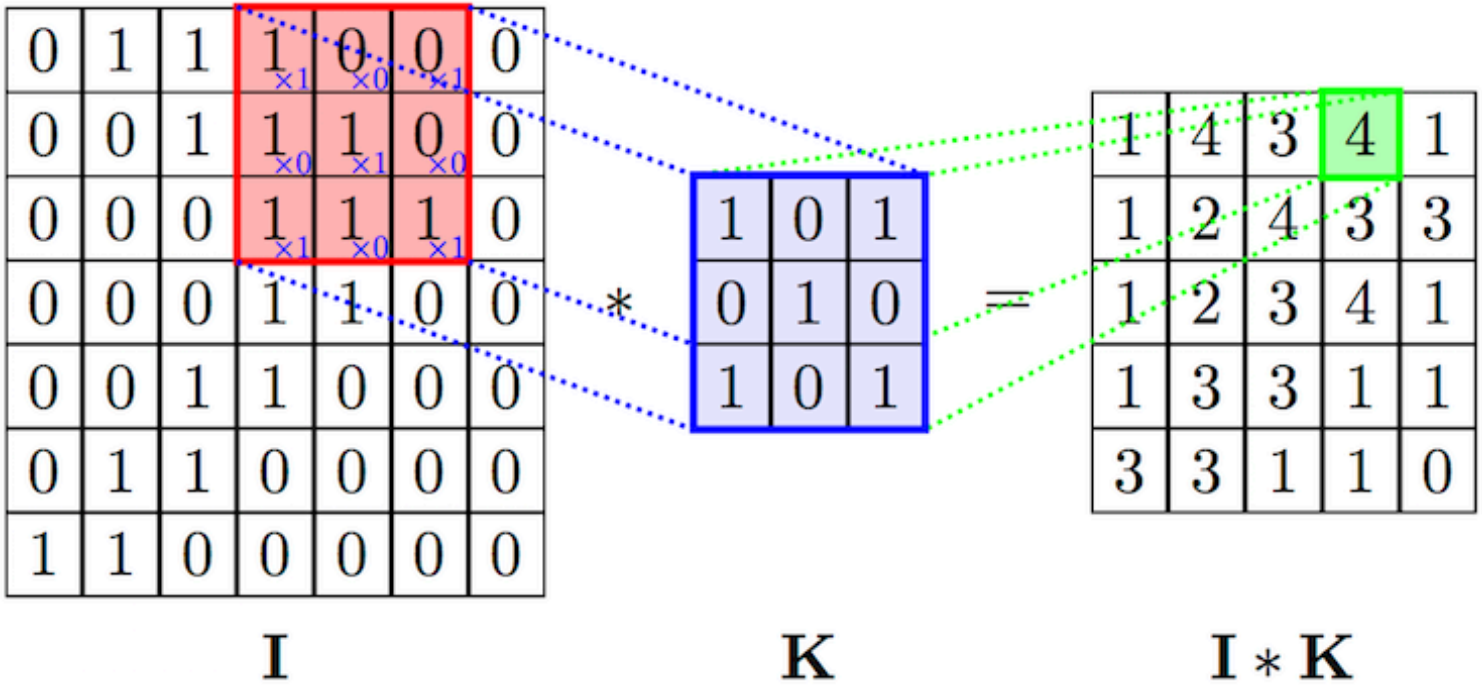
Introduction

- Artifical Intelligence
- > Robotics (embodied paradigme)
- > Developmental robotics
- > Multimodal learning
- > Gesture and Audio-Visual recogntion

Deep Network Architectures – Convolutional Neural Networks (CNNs)



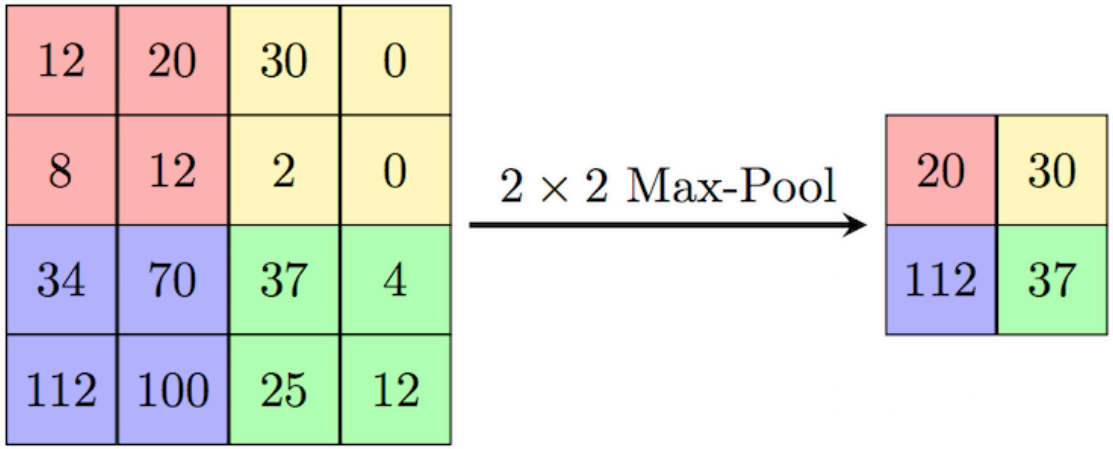
Deep Network Architectures – Convolutional Neural Networks (CNNs) – Convolution



Source: https://cambridgespark.com/content/tutorials/convolutional-neural-networks-with-keras/index.html

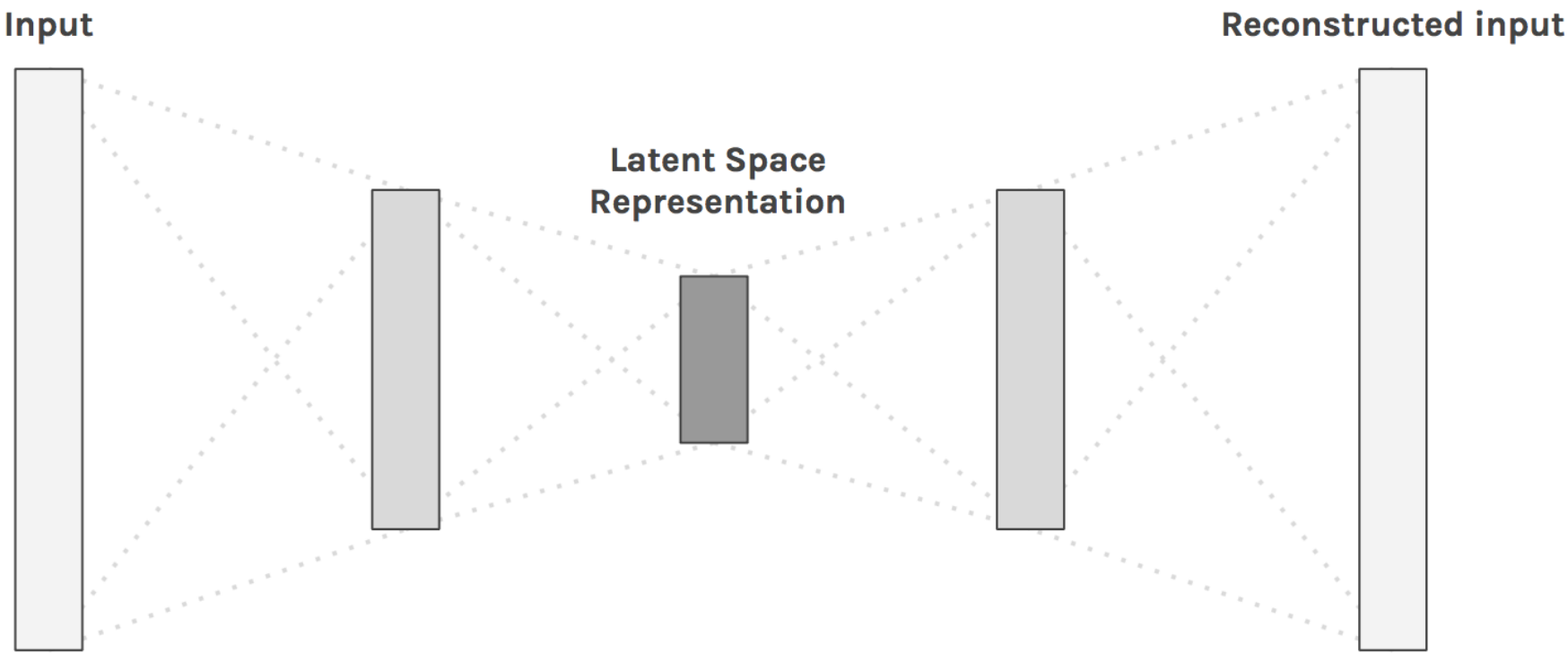
Also see https://github.com/vdumoulin/conv_arithmetic

Deep Network Architectures – Convolutional Neural Networks (CNNs) – Max-pooling



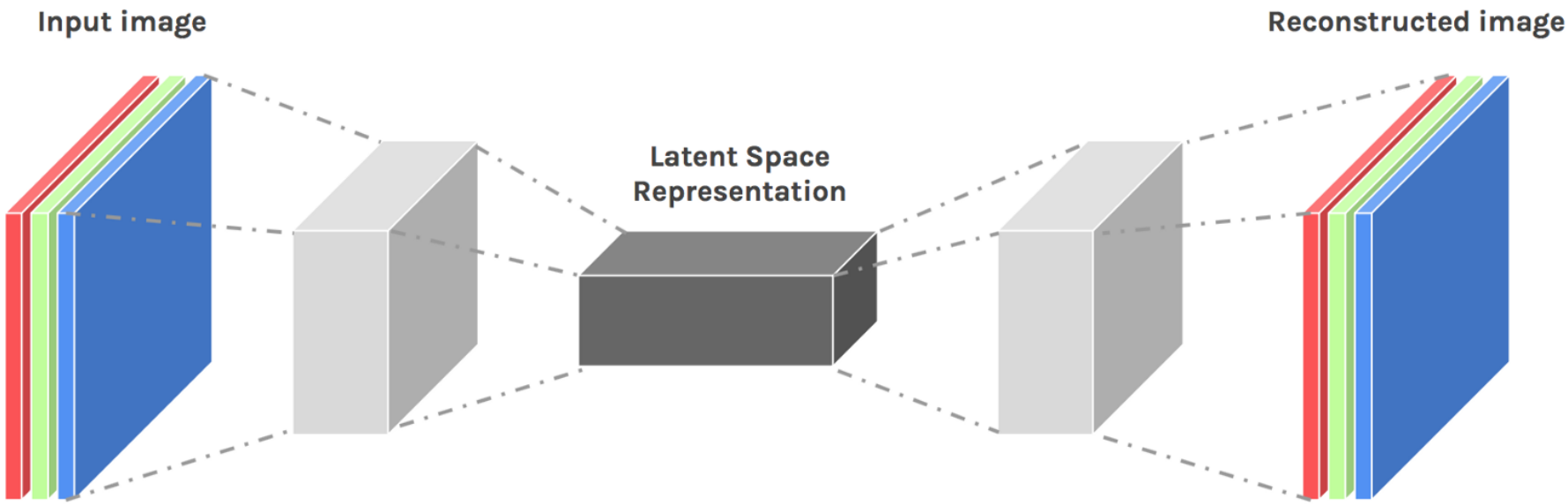
Source: https://cambridgespark.com/content/tutorials/convolutional-neural-networks-with-keras/index.html

Deep Network Architectures – Autoencoder



Source:
https://hackernoon.com/autoencoders-deep-learning-bits-1-11731e200694

Deep Network Architectures – Convolutional Autoencoder



Source:
https://hackernoon.com/autoencoders-deep-learning-bits-1-11731e200694

Training a Machine Learning Model

- Loss function: cross-entropy, L2-distance
- Stochastic gradient descent (SGD)
- Backpropagation
- Varaints of SGD: AdaGrad, Adam

Datasets – ASL Finger Spelling

- RGB and depth
- More than 60000 images for each modality
- 24 static signs in American Sign Language
- 5 subjects
- Only one channel in input
- Resized to 83×83 and Z-normalization

(a)

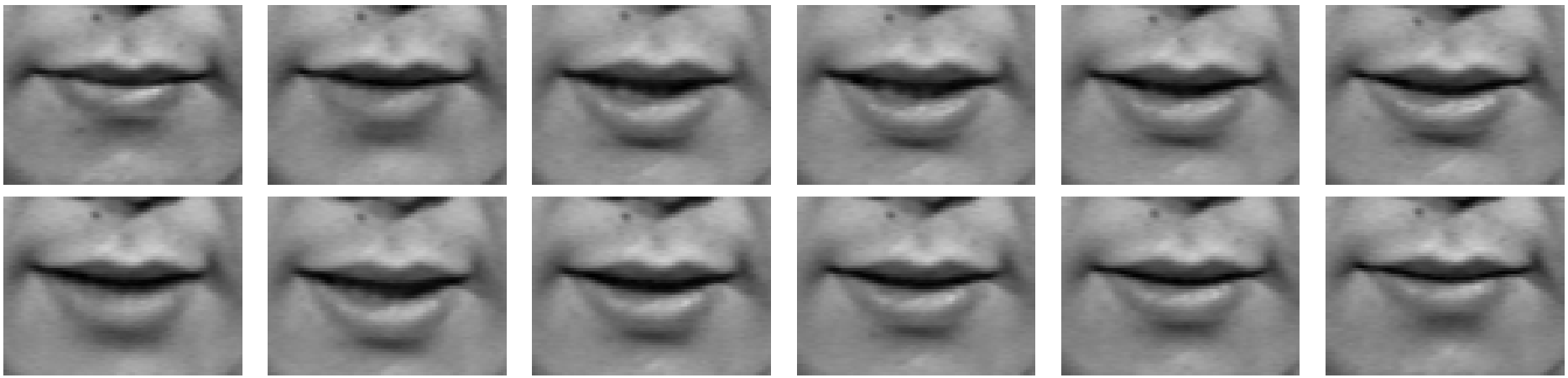
(b)

(c)

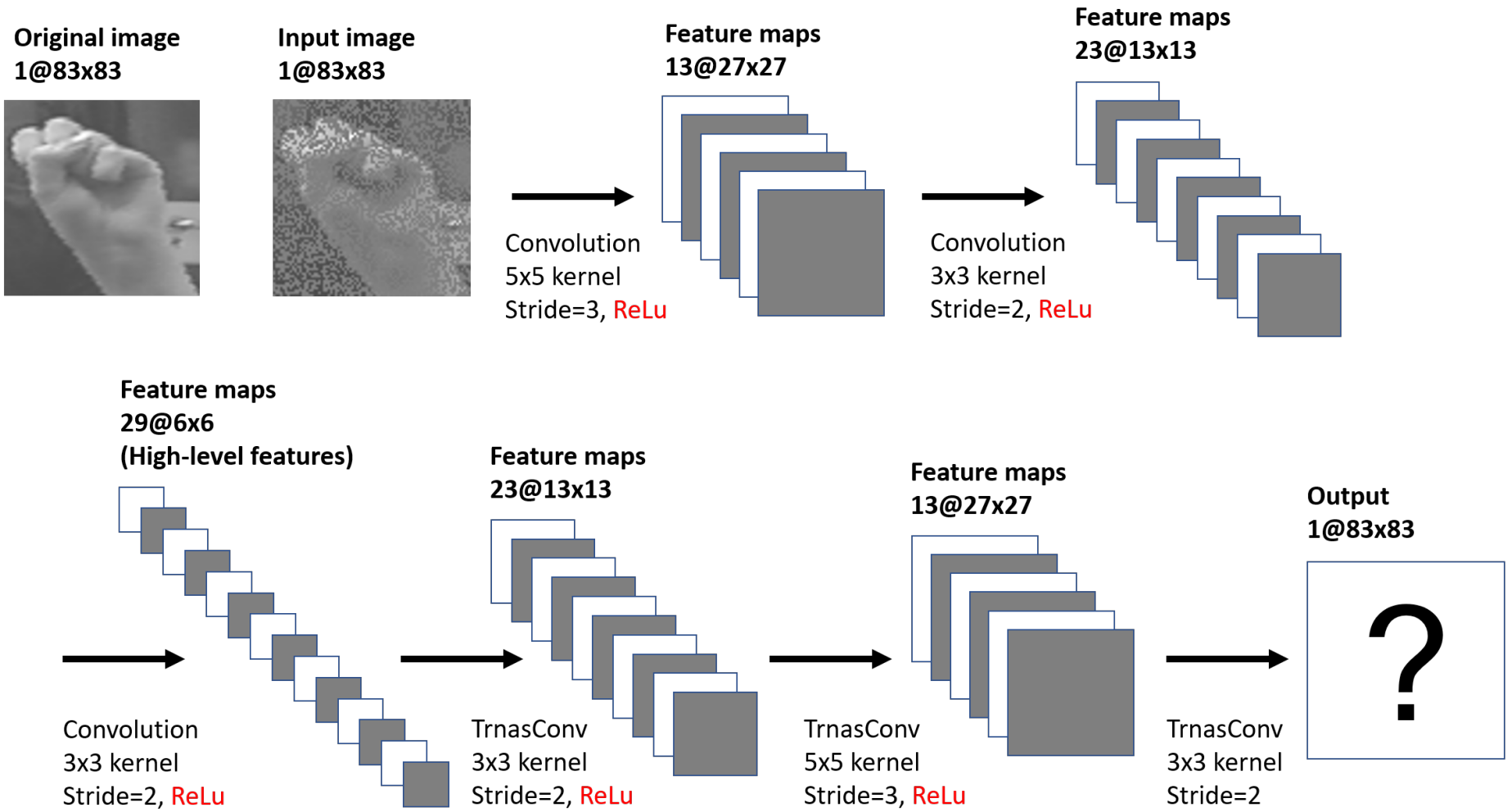
(d)

Datasets – AVLetters

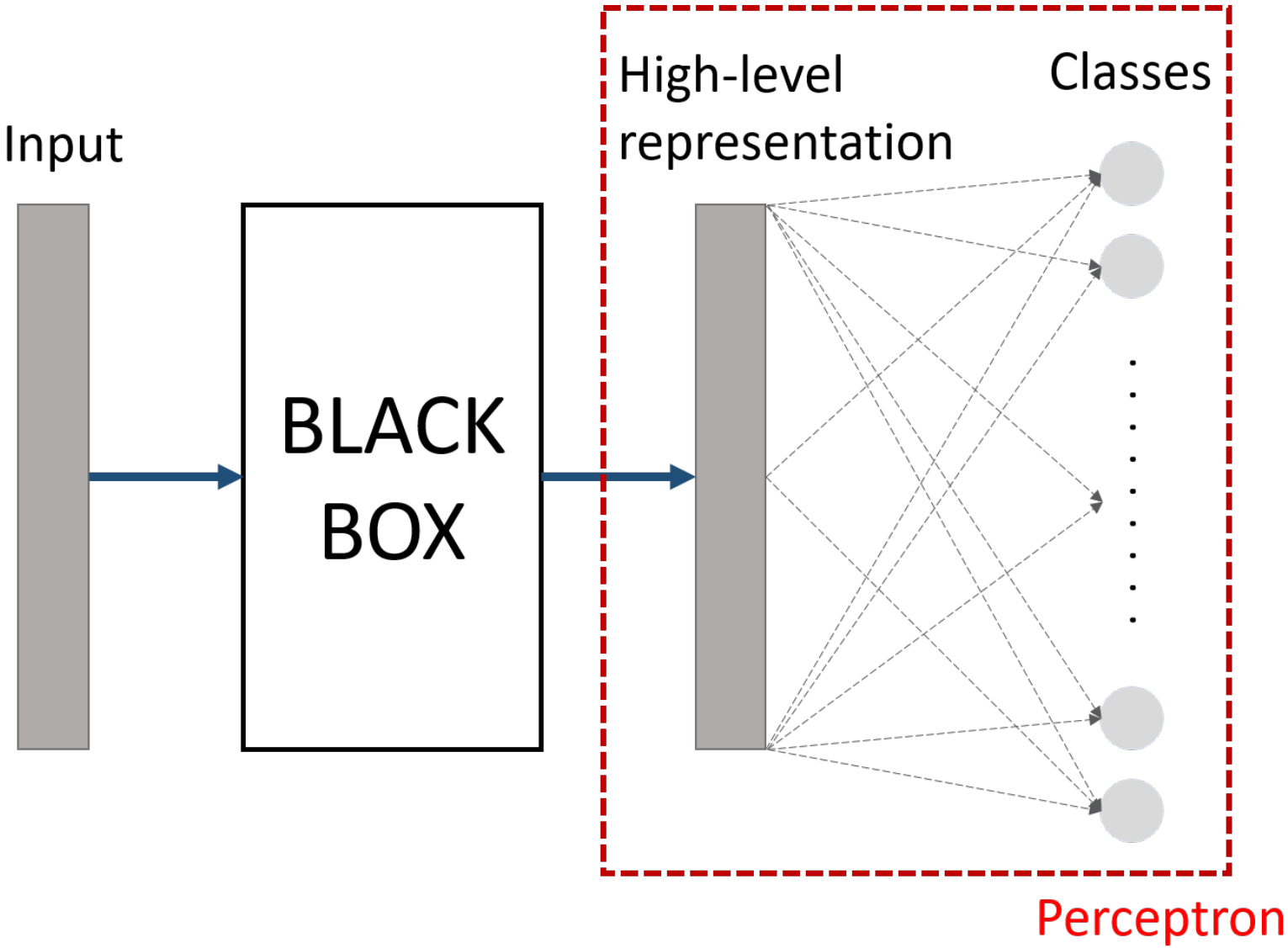
- Audio-visual
- 26 letters from A to Z
- 10 speakers, each letter 3 times each
- Audio: 24 frames in input, 26 MFCCs for each frame
- Video: 12 frames in input, z-normalized



Results – Unsupervised Learning with CAE



Results – Unsupervised Learning with CAE

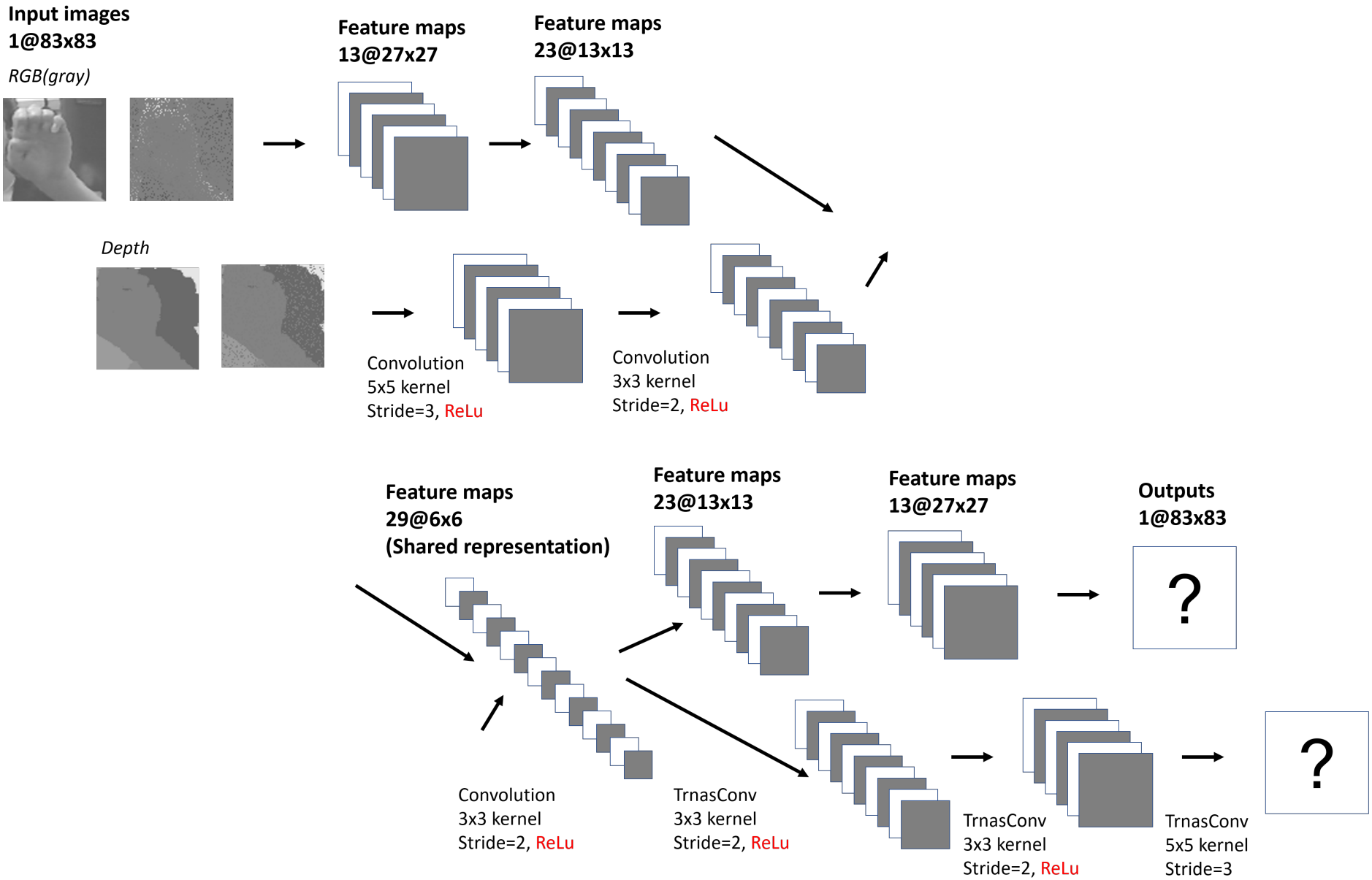


Results – Unsupervised Learning with CAE

- Raw: Perceptron that reads raw input data.
- CAE features: Perceptron stacked on the middle layer of the CAE.
- CAE architecture: Perceptron stacked on the middle layer of the CAE but train the whole network in a supervised way as a CNN.

		Raw	CAE features	CAE architecture
Intensity	train	69.47 %	78.87 %	91.29 %
	test	32.64 %	50.24 %	65.44 %
Depth	train	63.64 %	79.61 %	88.80 %
	test	29.93 %	41.64 %	55.62 %

Results – Shared Representation Learning

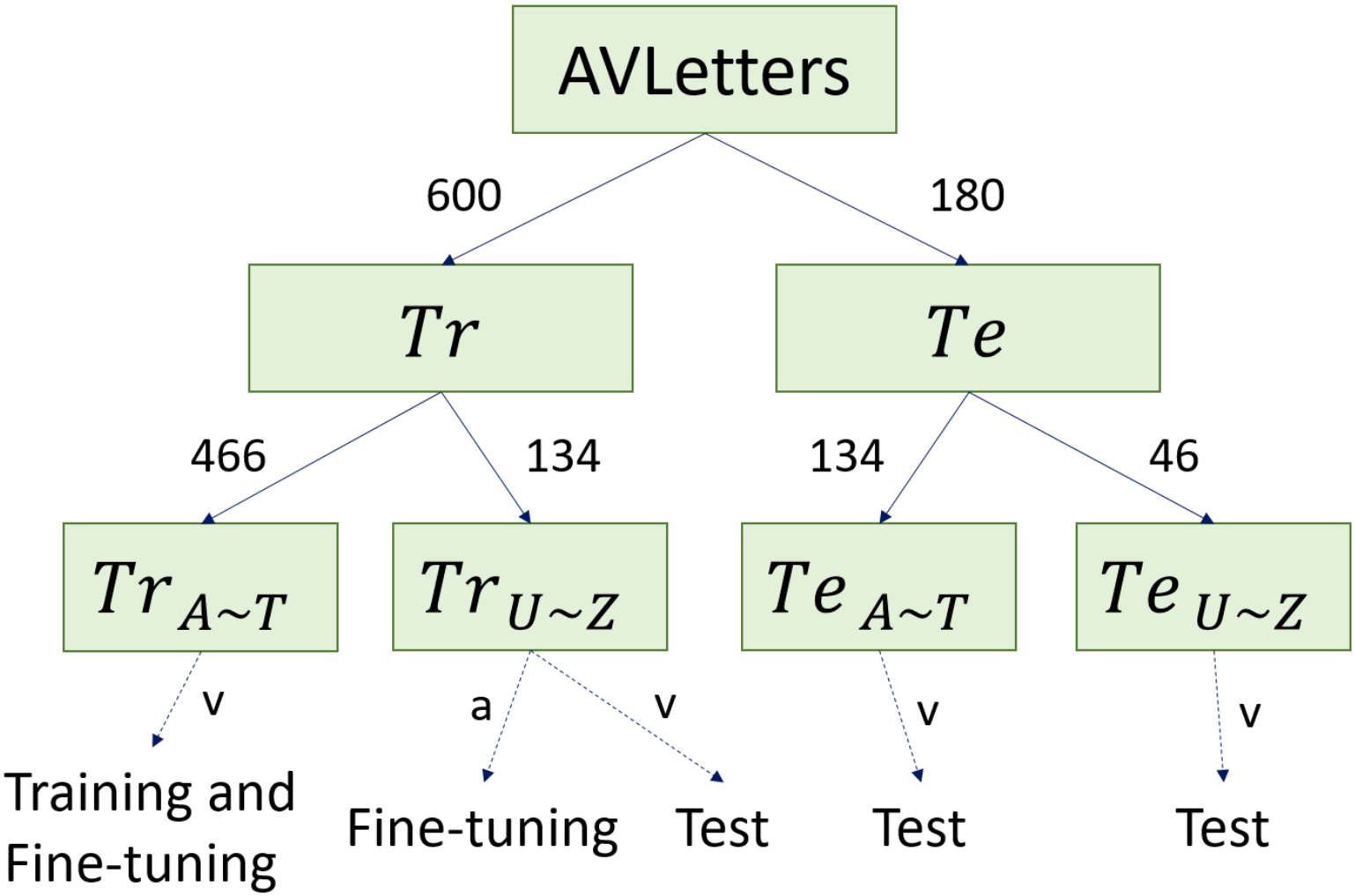


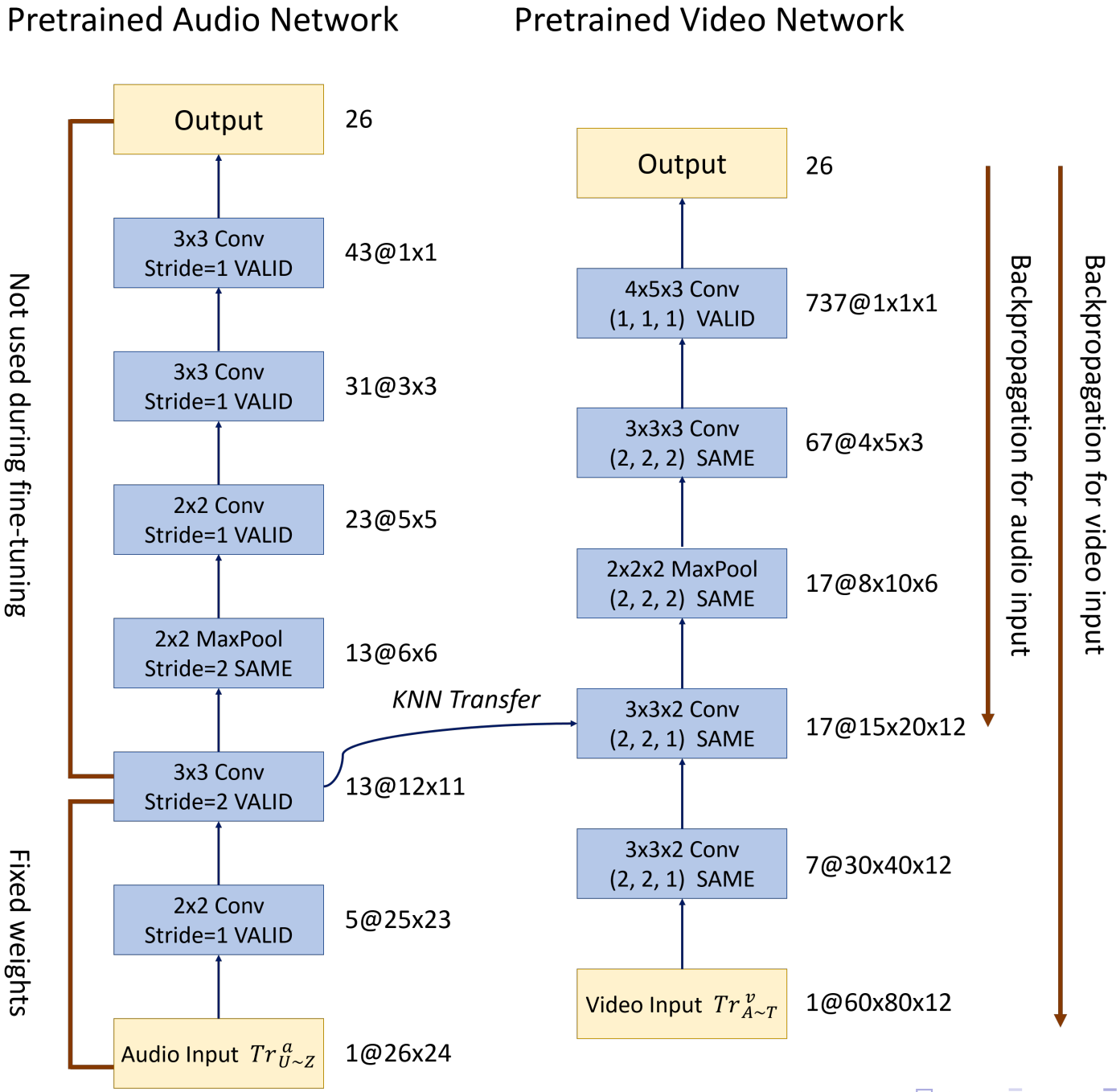
Results – Shared Representation

- Shared: Perceptron that exploits the shared representation learned by a bimodal CAE.

		Raw	CAE features	CAE architecture	Shared
Intensity	train	69.47 %	78.87 %	91.29 %	85.85 %
	test	32.64 %	50.24 %	65.44 %	53.38 %
Depth	train	63.64 %	79.61 %	88.80 %	81.83 %
	test	29.93 %	41.64 %	55.62 %	42.85 %

AVSR Knowledge Transfer




Pretrained Audio Network
Pretrained Video Network
Output
26
3x3 Conv
Stride=1 VALID
43@1x1
3x3 Conv
Stride=1 VALID
31@3x3
2x2 Conv
Stride=1 VALID
23@5x5
2x2 MaxPool
Stride=2 SAME
13@6x6
3x3 Conv
Stride=2 VALID
13@12x11
2x2 Conv
Stride=1 VALID
5@25x23
Audio Input $Tr^{a}_{U\sim Z}$
1@26x24
Not used during fine-tuning
Fixed weights
KNN Transfer
Output
26
4x5x3 Conv
(1, 1, 1) VALID
737@1x1x1
3x3x3 Conv
(2, 2, 2) SAME
67@4x5x3
2x2x2 MaxPool
(2, 2, 2) SAME
17@8x10x6
3x3x2 Conv
(2, 2, 1) SAME
17@15x20x12
3x3x2 Conv
(2, 2, 1) SAME
7@30x40x12
Video Input $Tr^{v}_{A\sim T}$
1@60x80x12
Backpropagation for audio input
Backpropagation for video input

AVSR Knowledge Transfer

Fine-tuned for 160 steps.
For **Exp1**, **Exp2** and **Exp3**, we have respectively $\alpha_0 = 0.001, 0.005, 0.001$ and $p_a = 0.85, 0.85, 1$.
Notice that since $p_a = 1$ for **Exp3** no video data is given in input during fine-tuning.

	Tr^v	$Tr^v_{A\sim T}$	$Tr^v_{U\sim Z}$	Te^v	$Te^v_{A\sim T}$	$Te^v_{U\sim Z}$
No transfer	77.67 %	**100** %	0 %	**40.56** %	**54.48** %	0 %
Exp1	**81.17** %	98.28 %	21.64 %	39.44 %	47.76 %	15.22 %
Exp2	40.83 %	51.07 %	5.22 %	23.89 %	30.60 %	4.35 %
Exp3	19.67 %	12.23 %	**45.52** %	12.22 %	2.24 %	**41.34** %

Conclusion

- Datasets, hyperparameters
- Applications in robotics
- Other approaches