# Learning disentangled representations of facial action sequences using convolutional encoding-decoding

Peter S. Li, Zhixin Shu, Dimitris Samaras

Computer Vision Lab, Stony Brook University

## Introduction

- Learning an interpretable manifold for visual data is a problem of interest in computer vision [1]
- Previous work has shown that it is possible to disentangle meaningful physical attributes using encoding-decoding [2]
- In this work, we explore learning a meaningful manifold for a disentangled sequence of facial actions (i.e. change from neutral expression to smiling) with a convolutional autoencoder, as well as how training with non-binary expression data affects the manifold
- The goal of training with intermediate expressions (between neutral and a full smile) is to make traversing the manifold smoother, improving performance in tasks such as interpolation
- We train the autoencoder to learn disentangling of expression and identity in facial images with binary expression data (smiling / neutral)
    - Uses a semi-supervised triplet training method

## Model Structure

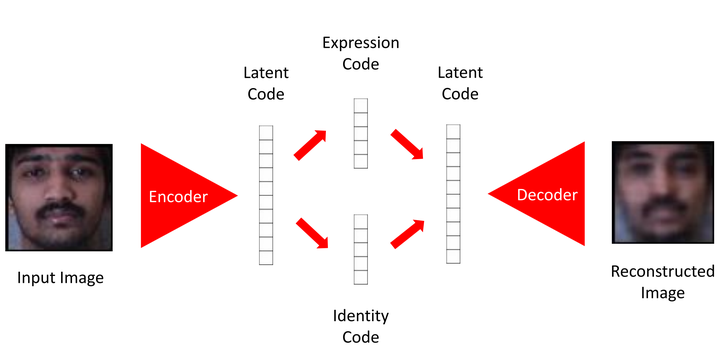


Figure 1: Illustration of the auto encoder and the separation of the latent code into expression and identity codes for disentangling

## Datasets and Data Processing

- For our experiments with binary disentangling, we trained our model with triplets of images from the Multi-PIE face database (Fig. 2)
- For training on non-binary expression intensities, we used videos from the MUG facial expression database
    - Videos were separated into image frames
- All images were tightly cropped around the face and resized to 64x64 pixels

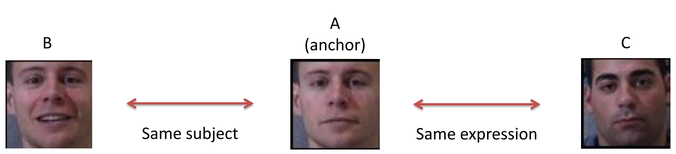


Figure 2: Example Multi-PIE image triplet

## Training Methods

- Standard autoencoder training: minimize mean squared error between input and reconstruction

Triplet training for binary disentanglement

- Encode each image into two 64-dimension latent vectors, and designate one vector to represent expression and one to represent identity (Fig. 1)
- Minimize cosine embedding, triplet, and L1 losses between latent vectors that should match based on their relationship within the image triplet
- Create an arbitrary label vector for each expression, and minimize a binary cross-entropy loss between the label vector and the expression vector
    - Label vectors used were [1,1,1,…] for smiling and [0,0,0,…] for neutral
- Minimize a “swapping loss” after decoding
    - In the latent space, swap expression vectors among images and then decode to form reconstructions
    - Minimize mean squared error between reconstructions and images they should ideally match in the dataset

Training with non-binary expression intensities from MUG

- Exposes the network to images with smile intensities between a neutral expression and a full smile
- Each MUG video frame is assigned a smile intensity
- Minimizes BCE loss on expression vector and an intensity label vector
    - Label vectors used were filled with values ranging from 0 to 1.0, incremented by 0.1

## Results

Figure 3: Expression swapping between subjects in Multi-PIE

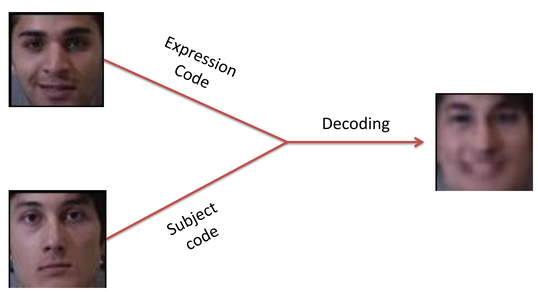


Figure 4: Interpolation between Multi-PIE test images after training with only binary expression data


## Discussion

- Swapping evaluation shows that binary disentangling was successful (Fig. 3)

Table 1: Tactics that improved disentangling

| Tactic | Reasoning |
|---|---|
| Tightly cropping all images around the face during data loading. | Less background pixels meant the network could allocate more resources to learning expression disentanglement. |
| Adding a swapping loss after the decoder. | Before adding the swapping loss, gradients were only being propagated back through the encoder. This meant that the decoder was unable to learn how to parse the encoder’s attempts at disentangling. The swapping loss ensures that gradients are passed back through the decoder so disentangling can be learned. |

- However, binary training is limited when it comes to the ultimate goal of this work: modeling change in expression as a linear movement on the manifold
- Interpolation between two faces is a good way to demonstrate this (Fig. 4)
    - The intermediate expressions that are generated by the model are lacking in realism
- The goal of training with the non-binary expression intensities from MUG is to smooth out the manifold, so that a linear traversal (like interpolation) will result in accurate intermediate expressions
- So far, interpolations generated by the MUG-trained model have been poor in visual quality, but we will continue exploring this moving forward

## Future Work

- In the future, the quality of our smiling manifold can be evaluated by comparing interpolated images to a real video of someone progressing from a neutral expression to smiling
- Another training strategy we could try is using a ranking based loss to train on the non-binary expression data
- To improve the quality of the reconstructions, we can use more complex architectures for the encoder and decoder
- This method of modeling action sequences with autoencoders can be applied more generally to many other actions
    - For example, modeling human and animal walking gaits

## Acknowledgements

This work was made possible with a grant from the Simons Foundation. Special thanks to David Rotunno, Anjalie Kini, and Helen Wang for the use of their computing resources.

## References

[1] Worrall, Daniel E., et al. “Interpretable Transformations with Encoder-Decoder Networks.” *ArXiv:1710.07307 [Cs]*, Oct. 2017. *arXiv.org*, http://arxiv.org/abs/1710.07307.

[2] Kulkarni, Tejas D., et al. “Deep Convolutional Inverse Graphics Network.” *ArXiv:1503.03167 [Cs]*, Mar. 2015. *arXiv.org*, http://arxiv.org/abs/1503.03167.