Deep Criminal Sketch Artist

CS 6955 Final Project - Fall 2020

Jadie Adams, Farshad Mogharrabi, and JC Zhu

Overview **2**

Dataset **2**

Network Architecture **3**

Approach 1 - Cross-Modal VAE **3**

Training 3

Results 5

Approach 2 - MVAE **6**

Training 6

Results 7

Comparison with Original MVAE **9**

Conclusion **10**

Code **10**

References **10**

Overview

Problem

When someone commits a crime and the victim remembers what the perpetrator looks like, but there is no photo evidence, a criminal sketch artist is used to render a drawing of the perpetrator based on the victim's description. Given the effectiveness of convolutional neural networks for image-based tasks, it is expected that this is a job that could be done via deep learning.

Proposed Solution

We Introduce “Deep Criminal Sketch Artist”, a multi-modal VAE model that can generate an image of a face given a description in the form of 40 binary attributes. This model makes use of a gaussian prior on the latent space which is shared by both the *image* modality and *feature* modality. By forcing the latent space to encode both images and attributes the model is able to produce more realistic images associated with the described attributes.

Motivation

A major benefit of a deep criminal sketch artist is that it would be faster and less expensive than a human criminal sketch artist. It could also be trained continuously to improve accuracy over time. For example, each time a criminal is caught, an image of their faces paired with the description from their victim could be added to the training data. In addition, iterating through multiple faces with the variation of features that the victim has no recollection of would be easier as the feature can be allowed to vary, and by looking at the variation of that resulting faces, the results can be evaluated and selected by the person.

Dataset

We used the CelebA dataset, which includes more than 200K celebrity images each with 40 attribute annotations. We used the aligned and cropped version of the images. The attribute annotations are formatted as a binary vector where each element encodes a feature such as:

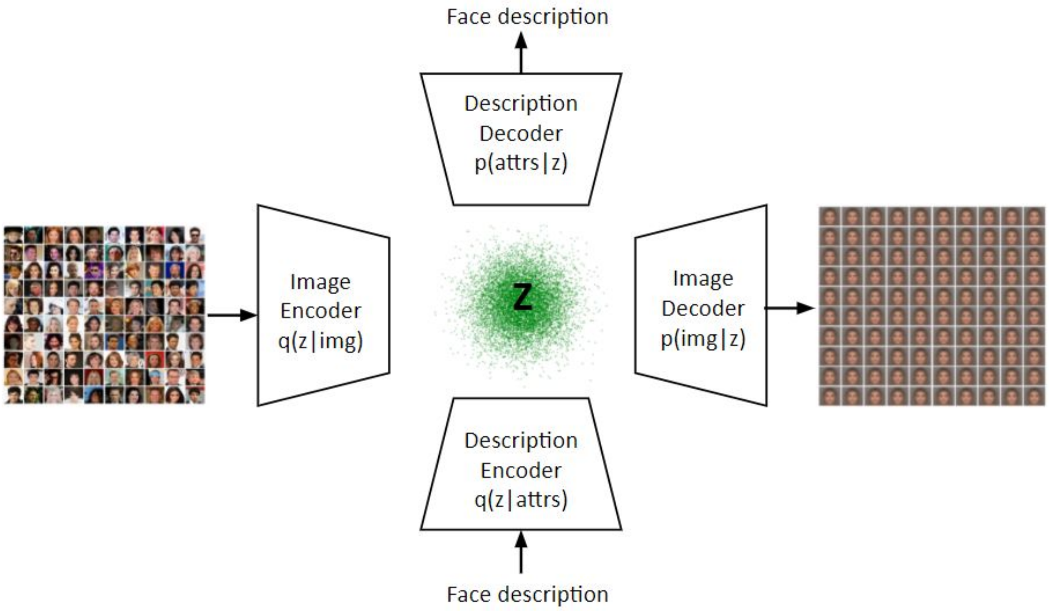
- 5_o_Clock_Shadow
- Arched_Eyebrows
- Attractive
- Bags_Under_Eyes
- Etc.

We re-sized the input images to be 64x64 to reduce the training time required. One issue we had was the CelebA dataset is too large to be automatically downloaded using torchvision in Google Collab. Because of this we needed to save the data locally and use a regular Jupyter Notebook or to use Google Collab the data needed to be copied to the google drive where the notebook was saved.

We split the data into three sets: training (162770 examples), validation (19867 examples), test (19962 examples) using the standard split listed in the CelebA file “list_eval_partition.txt”.

Network Architecture

The network is essentially composed of two VAEs: a convolutional VAE for the image modality (64x64 in dimension) and a fully connected VAE for the description or attribute modality (40 in dimension). We used 100 dimensions for the latent space, **Z**, and a spherical Gaussian standard normal as prior distribution.



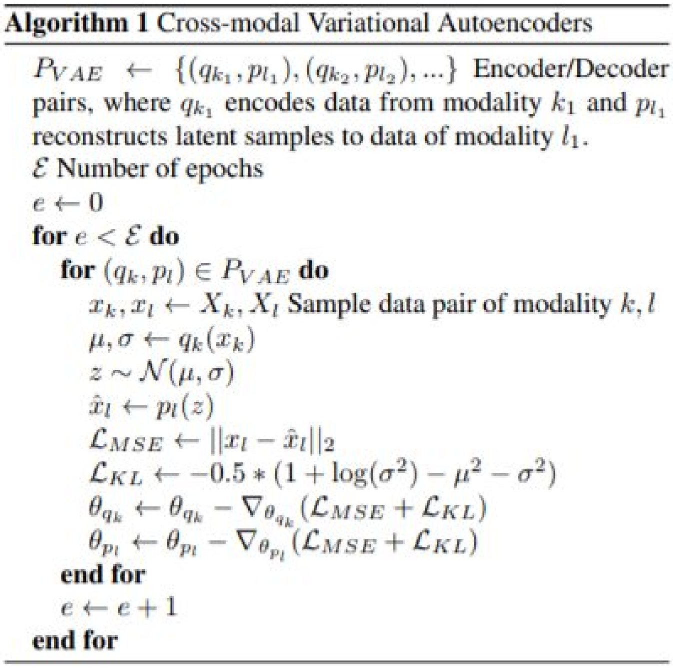
The encoders and decoders all had batch normalization and LeakyReLU activation. Both decoders had essentially the backwards architecture of their corresponding encoder. The image encoder architecture was based on the standard DCGAN architecture described here: Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. It has four convolutional layers (which grew in size) and two fully connected layers. The attribute encoder had three fully connected layers with hidden dimension size 512. Both encoders predict both the mean and log variance in the **z** space.

Approach 1 - Cross-Modal VAE

To train this network, we approached the problem in two different ways; the first approach was based on the paper Cross-Modal VAE for Hand Pose Estimation. This approach did not end up training very well, so we ended up trying another approach described later on.

Training

Below is the training algorithm from the paper which we implemented, where for us *k* is the attribute modality and *l* is the image modality.



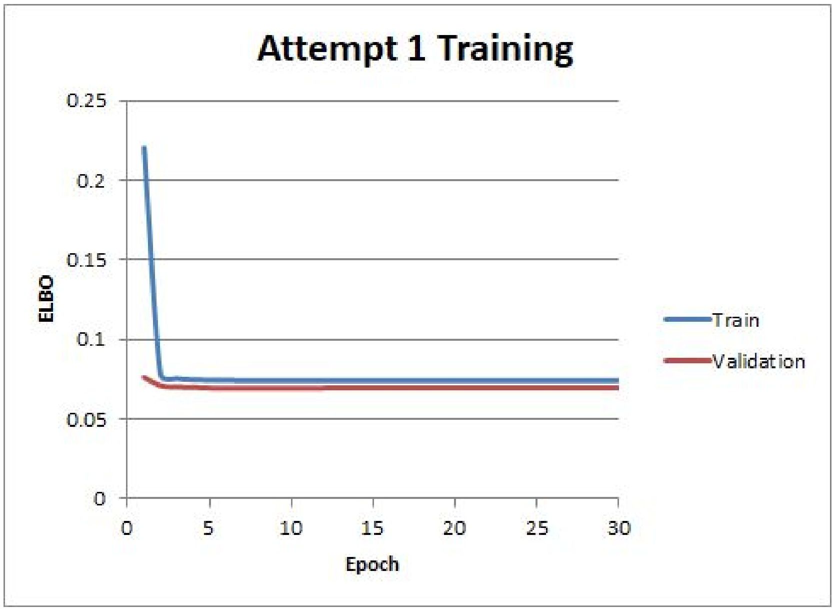
Algorithm 1 Cross-modal Variational Autoencoders

```
$P_{VAE} \leftarrow \{(q_{k_1}, p_{l_1}), (q_{k_2}, p_{l_2}), ...\}$ Encoder/Decoder pairs, where $q_{k_1}$ encodes data from modality $k_1$ and $p_{l_1}$ reconstructs latent samples to data of modality $l_1$.
$\mathcal{E}$ Number of epochs
$e \leftarrow 0$
for $e < \mathcal{E}$ do
    for $(q_k, p_l) \in P_{VAE}$ do
        $x_k, x_l \leftarrow X_k, X_l$ Sample data pair of modality $k, l$
        $\mu, \sigma \leftarrow q_k(x_k)$
        $z \sim \mathcal{N}(\mu, \sigma)$
        $\hat{x}_l \leftarrow p_l(z)$
        $\mathcal{L}_{MSE} \leftarrow ||x_l - \hat{x}_l||_2$
        $\mathcal{L}_{KL} \leftarrow -0.5 * (1 + \log(\sigma^2) - \mu^2 - \sigma^2)$
        $\theta_{q_k} \leftarrow \theta_{q_k} - \nabla_{\theta_{q_k}}(\mathcal{L}_{MSE} + \mathcal{L}_{KL})$
        $\theta_{p_l} \leftarrow \theta_{p_l} - \nabla_{\theta_{p_l}}(\mathcal{L}_{MSE} + \mathcal{L}_{KL})$
    end for
    $e \leftarrow e + 1$
end for
```

Essentially the attributes are encoded into a mean and variance in the **Z** space using the attribute encoder branch. The z is sampled from a normal distribution using this mean and variance (sampling z rather than using the mean as z makes it more robust to noise). This z then goes through the image decoder to get a “reconstructed” image.

The loss is the same ELBO loss used in regular VAEs, consisting of a reconstruction term (mean square error between the original and reconstructed image) and the KL divergence (which encourages the **Z** space to match the prior). The loss was applied to the entire network in backpropagation. This approach makes sense considering our task is to reconstruct the image given only the attributes.

We used a batch size of 100 in training. Below is a plot of how the ELBO loss for the training and validation sets changed over the epochs.



Results

As can be seen in the loss plot, the training quickly converged and the network stopped learning anything. What happened is the network learned to predict the same essentially featureless blurry face no matter the attributes. Essentially it was ignoring the attributes and learned to predict the average of all of the face images in the training set.

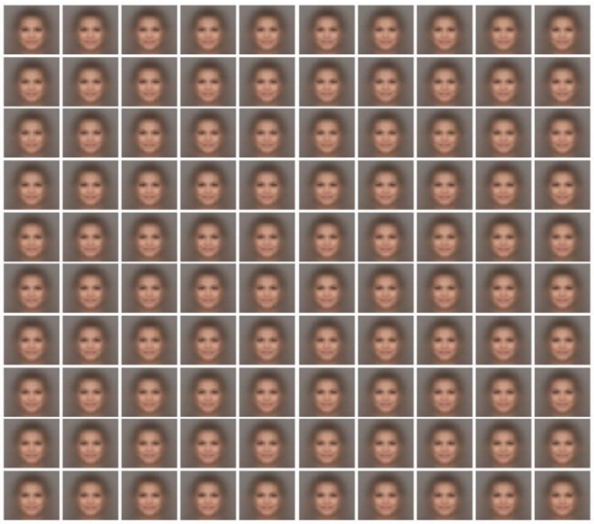
Test MSE

To quantify the success of our attempts, we consider the mean square error (MSE) between the real images and the images predicted using only the attributes corresponding to the real images on our unseen test set. The average MSE on the test set was 0.0750.

Image Reconstruction

Here, in the following figure, you can see an input batch of true face images and the face image predicted by the network using the corresponding attributes. As seen, the network has reconstructed the average face regardless of the features in the given images.

Real Images



Predicted Images

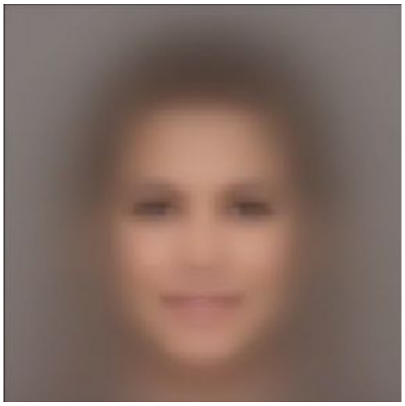
Sketch Test

To analyze our model’s success qualitatively, we defined three criminals using a few features and had our model produce a “sketch” of these criminals.

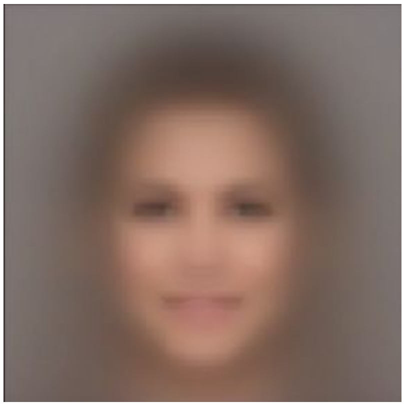
Criminal Defining Features:

- Criminal 1 - Man with gray hair, pale skin, and a mustache.
- Criminal 2 - Young woman with black hair, heavy makeup, wearing a hat.
- Criminal 3 - Chubby young man with glasses and bushy eyebrows.

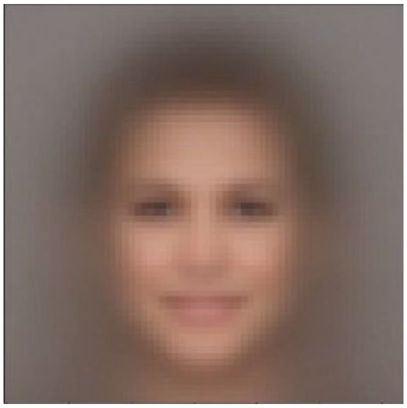
Criminal Sketches:



Criminal 1



Criminal 2



Criminal 3

Qualitatively we can see our deep criminal sketch artist thinks all criminals look about the same and failed at representing the provided feature.

Approach 2 - MVAE

Our second approach was based on this paper Multimodal Generative Models for Scalable Weakly-Supervised Learning following their open-source implementation https://github.com/mhw32/multimodal-vae-public.

This work defines a multi-modal VAE (MVAE) network with a shared latent space the same way, except it adds a product-of-experts inference network and a slightly more complex training technique. The product-of-experts combines the predictions of multiple probabilistic models, so in our case the product-of-experts combines the z prediction from the image encoder and attribute encoder to get a new z for decoding. The idea is by allowing the model to learn from both modalities combined, the model makes better predictions when one modality is missing.

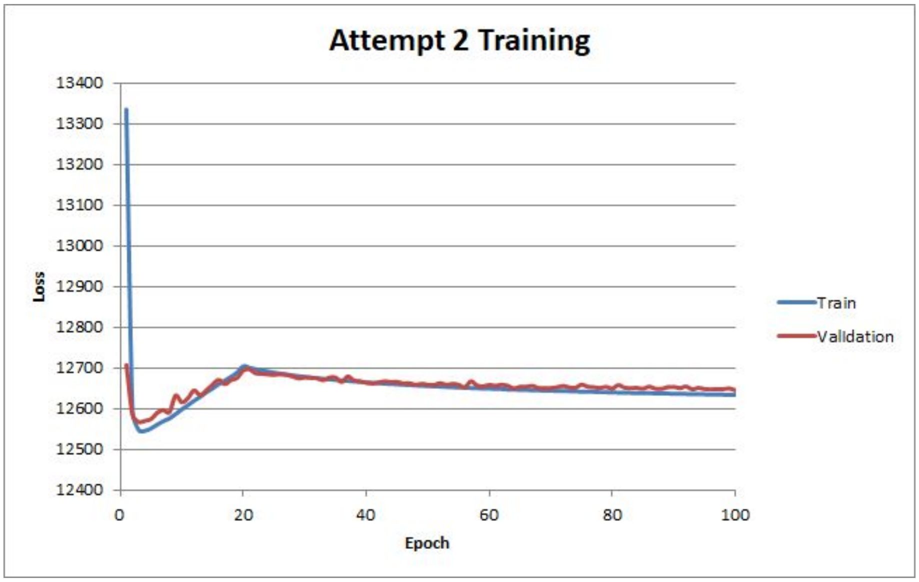
For this approach we left the existing architecture the same except we replaced the activation function with the one used in MVAE. They use the product of x and sigmoid(x) for the activation as suggested in: Searching for Activation Functions.

Training

The MVAE network loss consists of the sum of three factors: joint ELBO, image ELBO, and attribute ELBO. In training, initially both the image and attributes is encoded in the z space and the product-of-experts is used to get a single mean and variance. This mean and variance are used to get the KL-divergence term in the joint ELBO. Both the image and attributes are then reconstructed and the reconstruction term of the joint ELBO is the sum of the mean square errors between the original and predicted images and attributes. The image ELBO is found similarly, but only the images are encoded and likewise, for the attribute loss, only the attributes are encoded.

We also added an annealing factor as done in the MVAE work so that the KL divergence term is weighted less heavily in the early epochs. We set it so it would gradually increase in weight over the first 20 epochs then be set to one for all following. This allows the model to focus on learning the data before requiring it to match the prior in the latent space.

The training and validation losses are plotted below over epochs.



As you can see, the loss drops quickly when the KL-divergence term has a small weight, but as the weight increases and latent space is required to be Gaussian, the loss increases. After the 20th epoch, the annealing factor stays constant at one and we can see the loss slowly decrease as we would expect.

Results

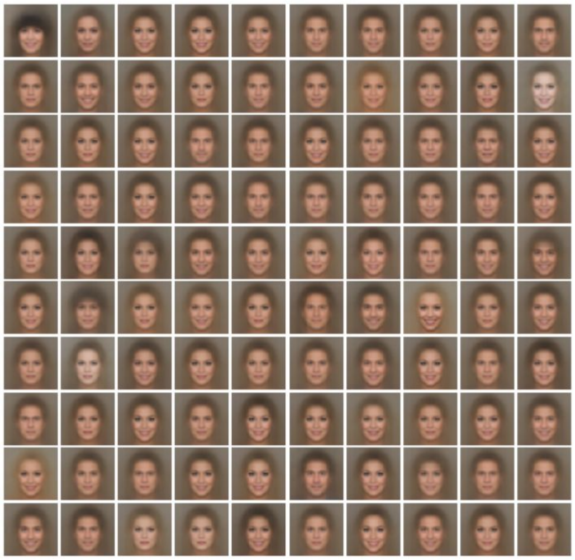
Test MSE

The average MSE between the real images and images predicted using only the attributes was 0.7804 on the test set. This is larger than the previous model, but as the qualitative results will show, MSE is not a great measure of accuracy for this task and this model did do better.

Image Reconstruction

Below you can see an input batch of true face images and the face image predicted by the network using the corresponding attributes.

Real Images



Predicted Images

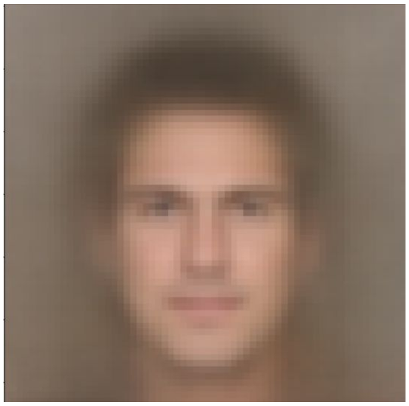
Looking closely at these images, we can see that the model successfully represented some of the major features like hair color and gender. We can also see that it did not attempt to reconstruct noise in the image, such as the background.

Sketch Test
Below are the sketches this model made from our criminal descriptions.

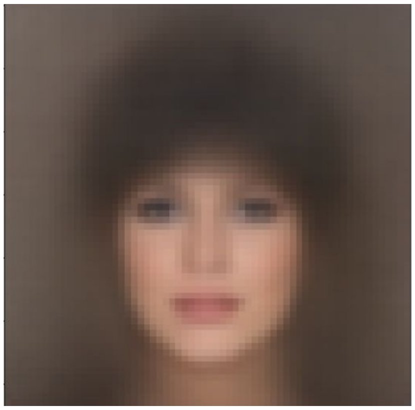
Criminal Defining Features:

- Criminal 1 - Man with gray hair, pale skin, and a mustache.
- Criminal 2 - Young woman with black hair, heavy makeup, wearing a hat.
- Criminal 3 - Chubby young man with glasses and bushy eyebrows.

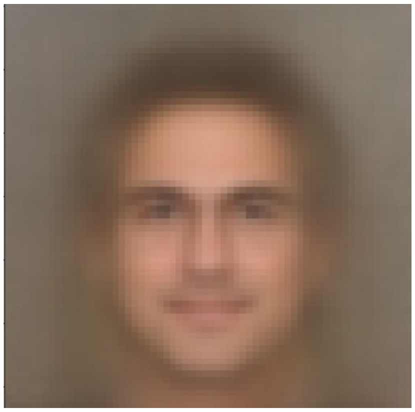
Criminal Sketches:



Criminal 1



Criminal 2



Criminal 3

Our model successfully captured some features. For example the gender and hair color look correct and criminal 3 who was described as “chubby” has a wider face than criminal 1. However, many features are missing, such as criminal one’s mustache and criminal three’s glasses. It is likely that these features are somewhat uncommon in the CelebA dataset and so the model did not learn to represent them as well.

Comparison with Original MVAE

We were interested to see whether our implementation of MVAE was worse than the paper author’s implementation so we tried our sketch test on their model from this repo: https://github.com/mhw32/multimodal-vae-public

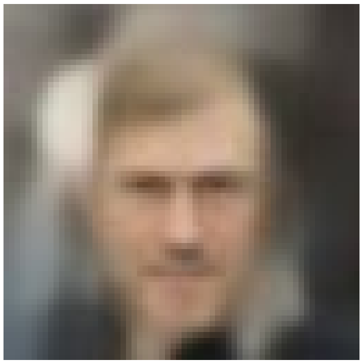
Sketch Test

Below are the sketches their model made from our criminal descriptions.

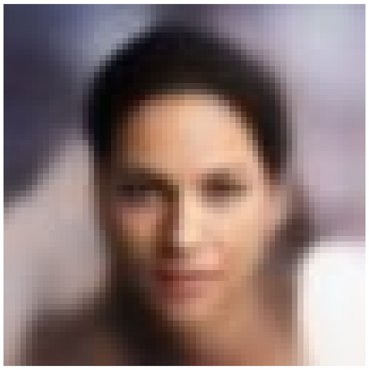
Criminal Definitions:

- Criminal 1 - Man with gray hair, pale skin, and a mustache.
- Criminal 2 - Young woman with black hair, heavy makeup, wearing a hat.
- Criminal 3 - Chubby young man with glasses and bushy eyebrows.

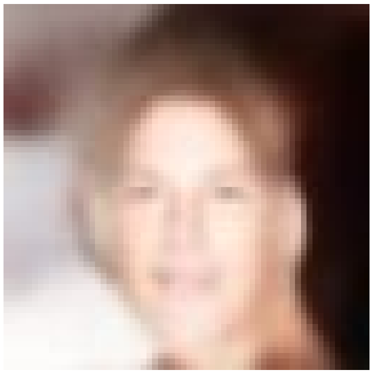
Criminal Sketches:



Criminal 1



Criminal 2



Criminal 3

Their sketches look a bit more like realistic faces and even some background and they capture some of the features better. For example, you can see a slight mustache on Criminal 1. However, their model missed some of the same features ours did like the eyeglasses on Criminal 3.

The main difference in our implementation from the original MVAE is we used all 40 CelebA attributes whereas they select only 13 attributes. They likely chose the most statistically significant attributes as weeded out the ones which rarely appear. This most likely helped their model learn a bit better.

Conclusion

While we were somewhat successful in creating a deep criminal sketch artist, the predicted images lacked detail and excluded some of the given features. If we had time to improve this further, we might consider trying to balance the dataset so all of the attributes are well represented or remove attributes that are not well represented as was done in the original MVAE paper.

The addition of the product-of-experts and more complicated loss function did help our second approach to do better than our first.

Code

The implementation for our first approach is included in the "Approach1.ipynb" notebook and the implementation of the second approach is in the "Approach2.ipynb" notebook.

The code is self contained in the notebooks but does require downloading the CelebA dataset to run.

References

Our implementation was based on this repo:

https://github.com/mhw32/multimodal-vae-public

The ideas are based on these papers:

- Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks
- Cross-Modal VAE for Hand Pose Estimation
- Multimodal Generative Models for Scalable Weakly-Supervised Learning
- Generalized Product of Experts for Automatic and Principled Fusion of Gaussian Process Predictions