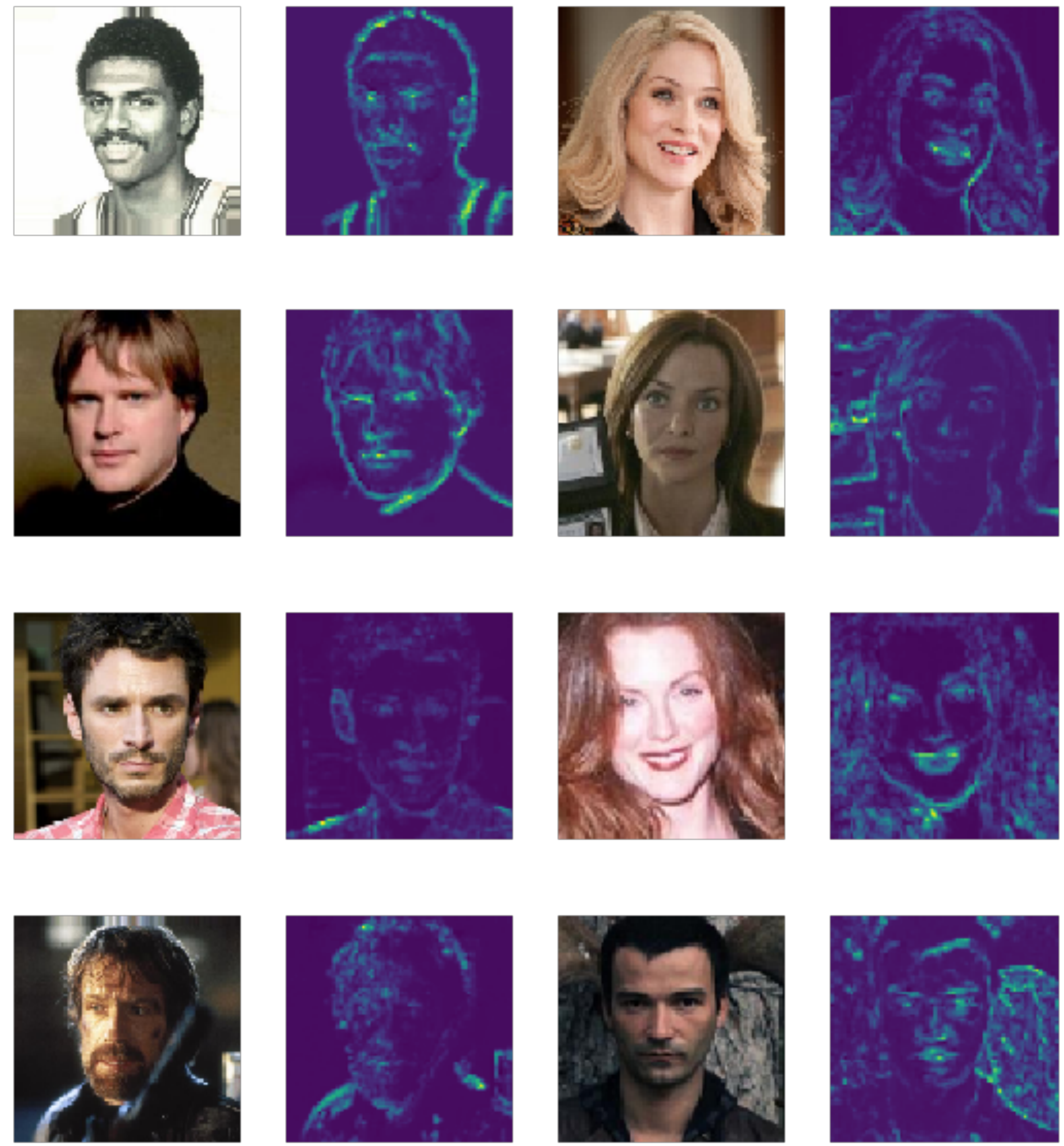
Face Image Analysis
Final Capstone

Supervised and Unsupervised Learning With Keras/TensorFlow

Max Calabro



Ghost Chuck Norris?

The Data:

~200,000 pre-cropped images from IMDB and Wikipedia with age metadata.

The Question:

- **Can we use these images alone to predict age?**
- **Can we automatically cluster these images into meaningful groups?**

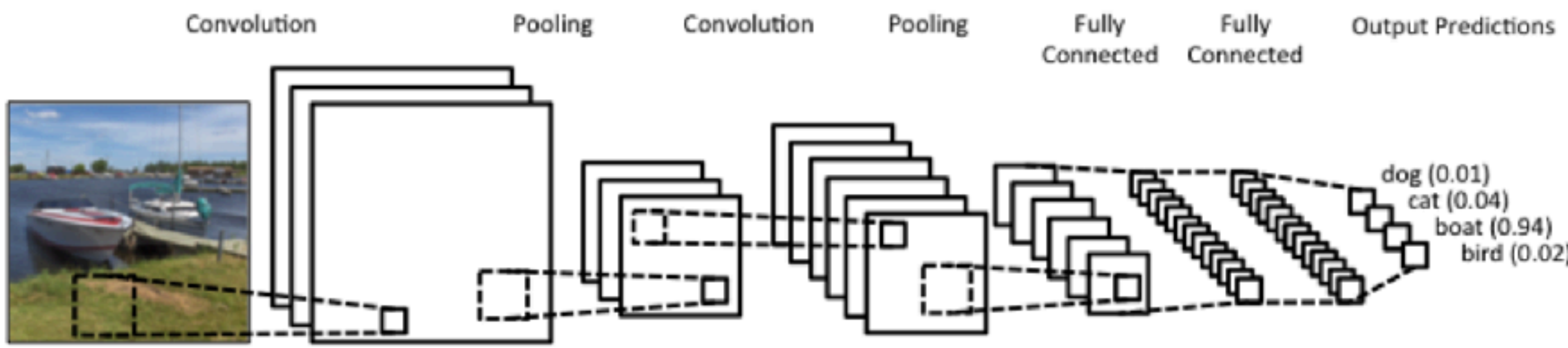
Overview

1. Preprocessing Steps
2. What are Convolutional Neural Networks?
3. Supervised Model - Predict Age
4. “Autoencoder” and “Attention Map”
5. Unsupervised Modeling - Clustering
6. Back to the Supervised Model
7. Conclusion

Data Preprocessing

1. Remove images with multiple faces.
2. Remove images with low “face score”.
3. Remove age outliers (< 15, > 65).
4. Remove small images (< 90 x 90 pixels).
5. Split the data into training (175,000) and validation (10,000).
6. Resample training set to balance ages.
7. Resize (150 x 150 pixels) and normalize images.
8. Split into batches for modeling.
9. Other complications: GPU, storage, memory, keras/tensorflow

Quick CNN Summary



- Convolutional layers convert image into “filters”.
- Can stack multiple filters on top of each other.
- Pooling layers take the maximum values from a filter layer, reducing dimensionality.
- Reduced dimension arrays get passed to standard “dense” (fully connected) neural network.
- Dense layers provide output in a useful format (predictions).

Age Prediction

My Supervised Model:

```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 148, 148, 64)      1792
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 146, 146, 64)      36928
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 73, 73, 64)        0
_________________________________________________________________
dropout_6 (Dropout)          (None, 73, 73, 64)        0
_________________________________________________________________
flatten_4 (Flatten)          (None, 341056)            0
_________________________________________________________________
dense_9 (Dense)              (None, 128)               43655296
_________________________________________________________________
dropout_7 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_10 (Dense)             (None, 1)                 129
=================================================================
```

two filter layers

pooling layer

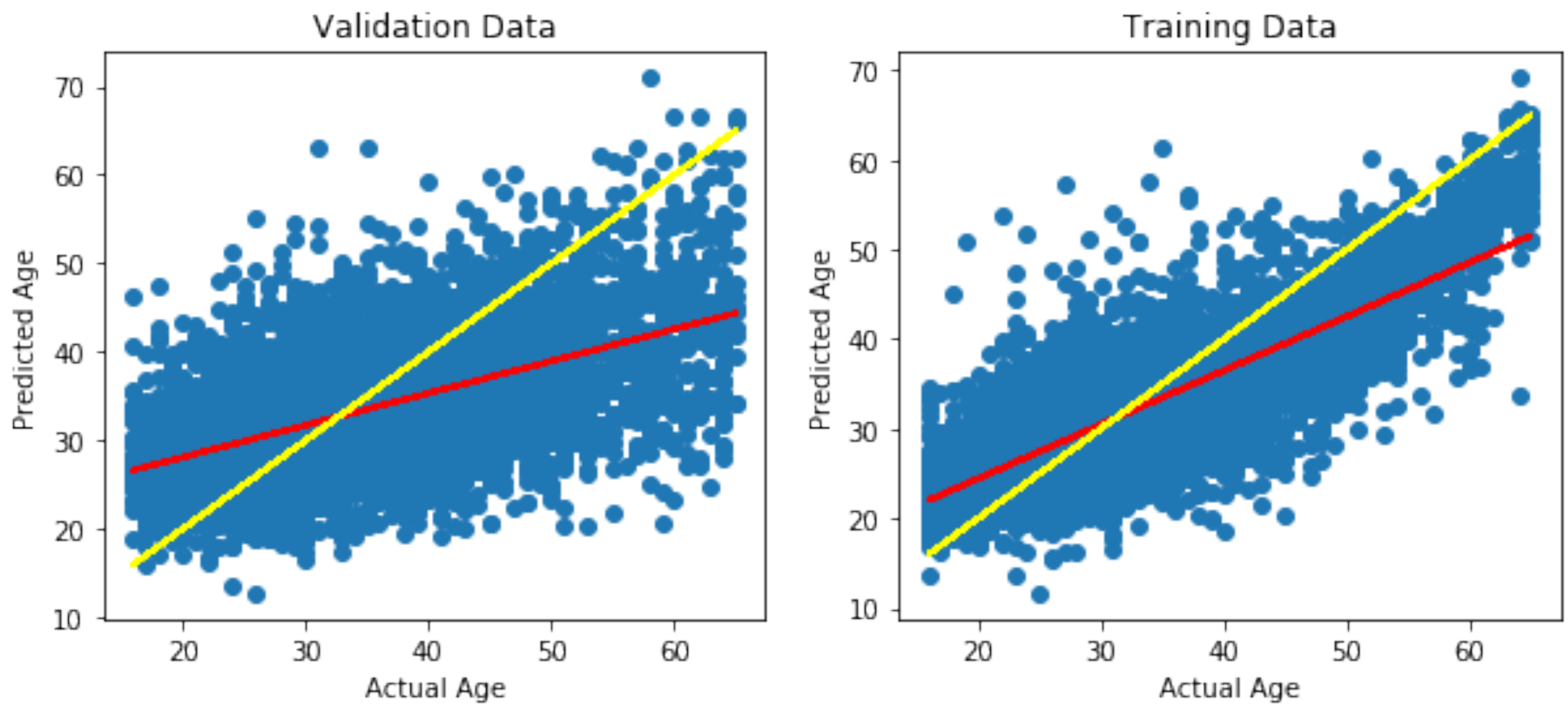
dropout to avoid overfitting

dense layers

one output: “age”

Ran 175,000 images through 10 epochs on a GPU.

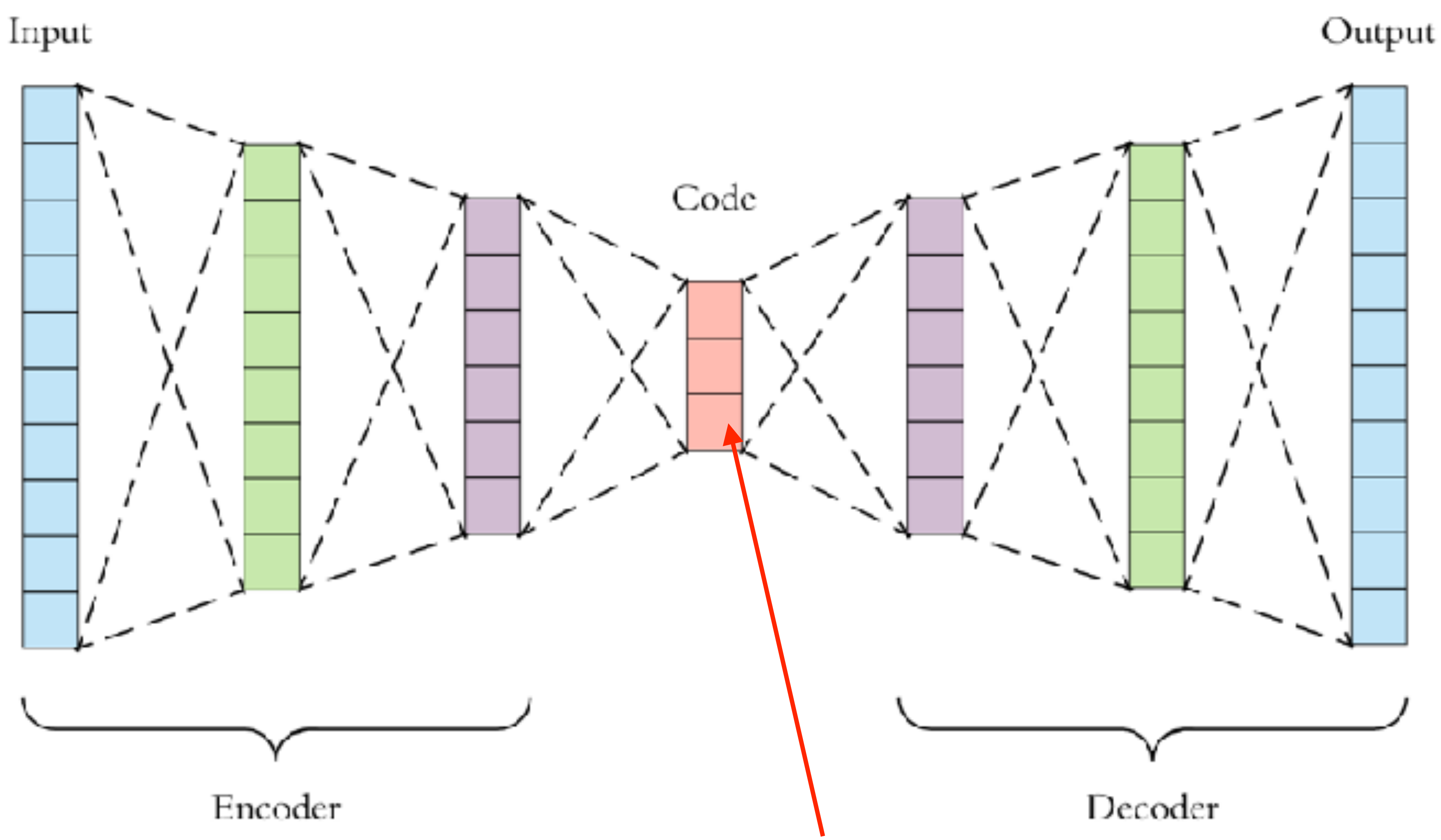
Age Prediction



- **Model performs poorly.**
- **Barely even going the right direction on validation.**
- **Overfitting training set.**
- **Predicting age is hard. Let's come back to this model later.**

Autoencoder

An autoencoder uses a CNN to reduce the dimensionality of an image by creating an internal layer with fewer nodes, then training with X_input = X_output.



We'll use this "compressed" information later.

Autoencoder

Encoding…

filter layers

pooling layers
(to downsample)

Compressed!

Decoding…

filter layers

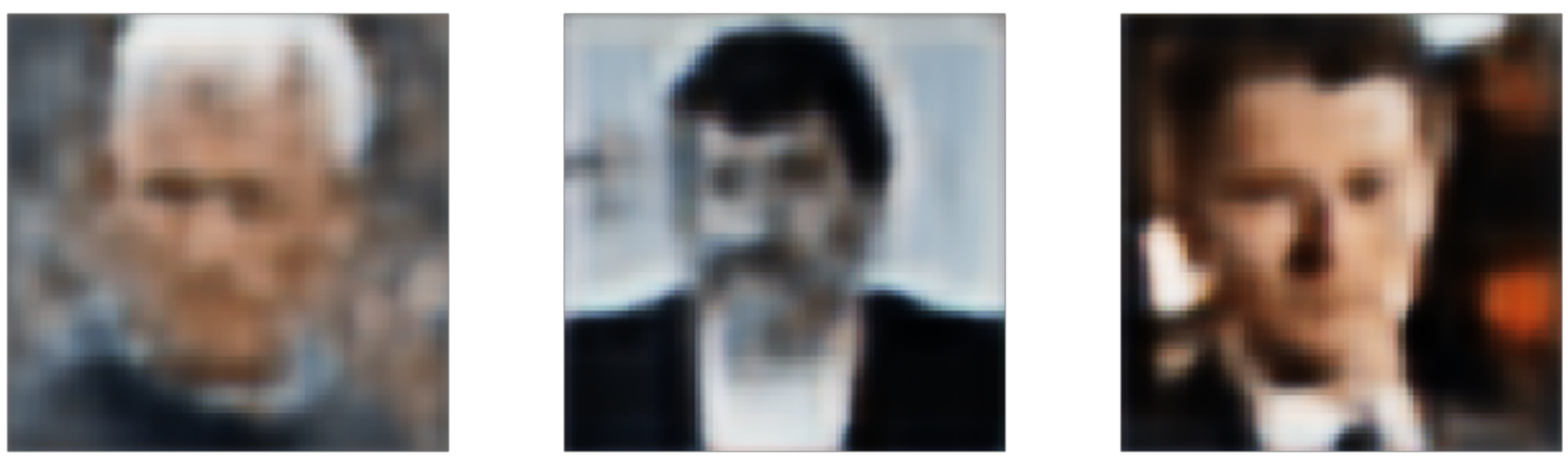
upsampling
layers

```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_8 (Conv2D)            (None, 150, 150, 32)      896
_________________________________________________________________
max_pooling2d_4 (MaxPooling2 (None, 75, 75, 32)        0
_________________________________________________________________
conv2d_9 (Conv2D)            (None, 75, 75, 16)        4624
_________________________________________________________________
max_pooling2d_5 (MaxPooling2 (None, 38, 38, 16)        0
_________________________________________________________________
conv2d_10 (Conv2D)           (None, 38, 38, 8)         1160
_________________________________________________________________
max_pooling2d_6 (MaxPooling2 (None, 19, 19, 8)         0
_________________________________________________________________
conv2d_11 (Conv2D)           (None, 19, 19, 8)         584
_________________________________________________________________
up_sampling2d_4 (UpSampling2 (None, 38, 38, 8)         0
_________________________________________________________________
conv2d_12 (Conv2D)           (None, 38, 38, 16)        1168
_________________________________________________________________
up_sampling2d_5 (UpSampling2 (None, 76, 76, 16)        0
_________________________________________________________________
conv2d_13 (Conv2D)           (None, 76, 76, 32)        4640
_________________________________________________________________
up_sampling2d_6 (UpSampling2 (None, 152, 152, 32)      0
_________________________________________________________________
conv2d_14 (Conv2D)           (None, 150, 150, 3)       867
=================================================================
```

Autoencoder

Before ENCODING:

After DECODING:



We lose information in the process, but can use the “compressed” values, which contain only the most important information.

Attention Map

Visualizing the internal layer of the autoencoder allows us to see which pixels are “important” to our model.

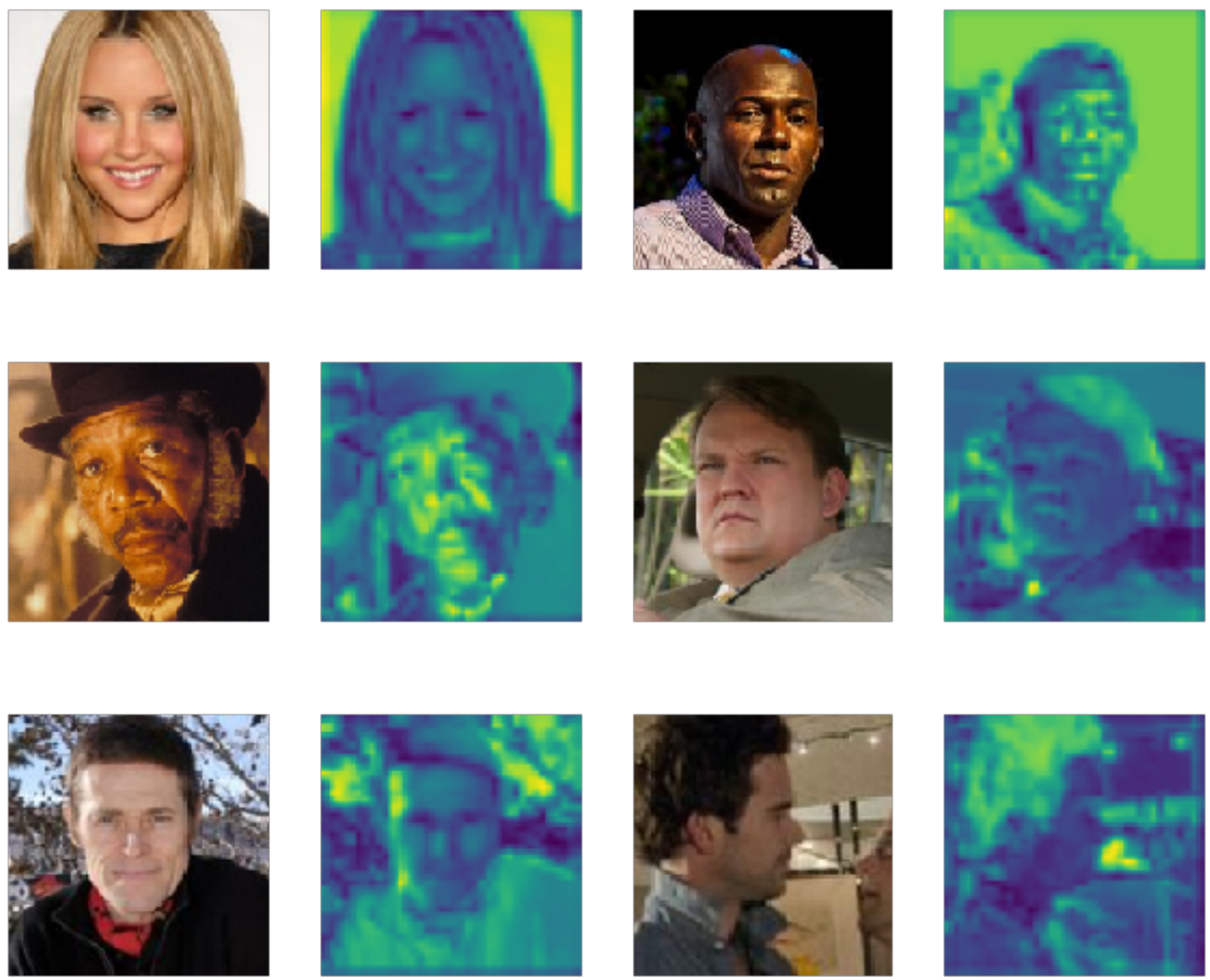
```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_8 (Conv2D)            (None, 150, 150, 32)      896
_________________________________________________________________
max_pooling2d_4 (MaxPooling2 (None, 75, 75, 32)        0
_________________________________________________________________
conv2d_9 (Conv2D)            (None, 75, 75, 16)        4624
_________________________________________________________________
max_pooling2d_5 (MaxPooling2 (None, 38, 38, 16)        0
_________________________________________________________________
conv2d_10 (Conv2D)           (None, 38, 38, 8)         1160
_________________________________________________________________
max_pooling2d_6 (MaxPooling2 (None, 19, 19, 8)         0
_________________________________________________________________
conv2d_11 (Conv2D)           (None, 19, 19, 8)         584
_________________________________________________________________
up_sampling2d_4 (UpSampling2 (None, 38, 38, 8)         0
_________________________________________________________________
conv2d_12 (Conv2D)           (None, 38, 38, 16)        1168
_________________________________________________________________
up_sampling2d_5 (UpSampling2 (None, 76, 76, 16)        0
_________________________________________________________________
conv2d_13 (Conv2D)           (None, 76, 76, 32)        4640
_________________________________________________________________
up_sampling2d_6 (UpSampling2 (None, 152, 152, 32)      0
_________________________________________________________________
conv2d_14 (Conv2D)           (None, 150, 150, 3)       867
=================================================================
```

Visualize this layer!

We’ll still have 19x19x8 = 2,888 features, but that’s better than 150x150x3 = 67,500!

Attention Map

This model doesn't care about faces any more than background pixels, so we don't expect the faces to be brighter.



Clustering

Used K-Means to cluster the “compressed” versions of the images (175,000 x 2,888 array).

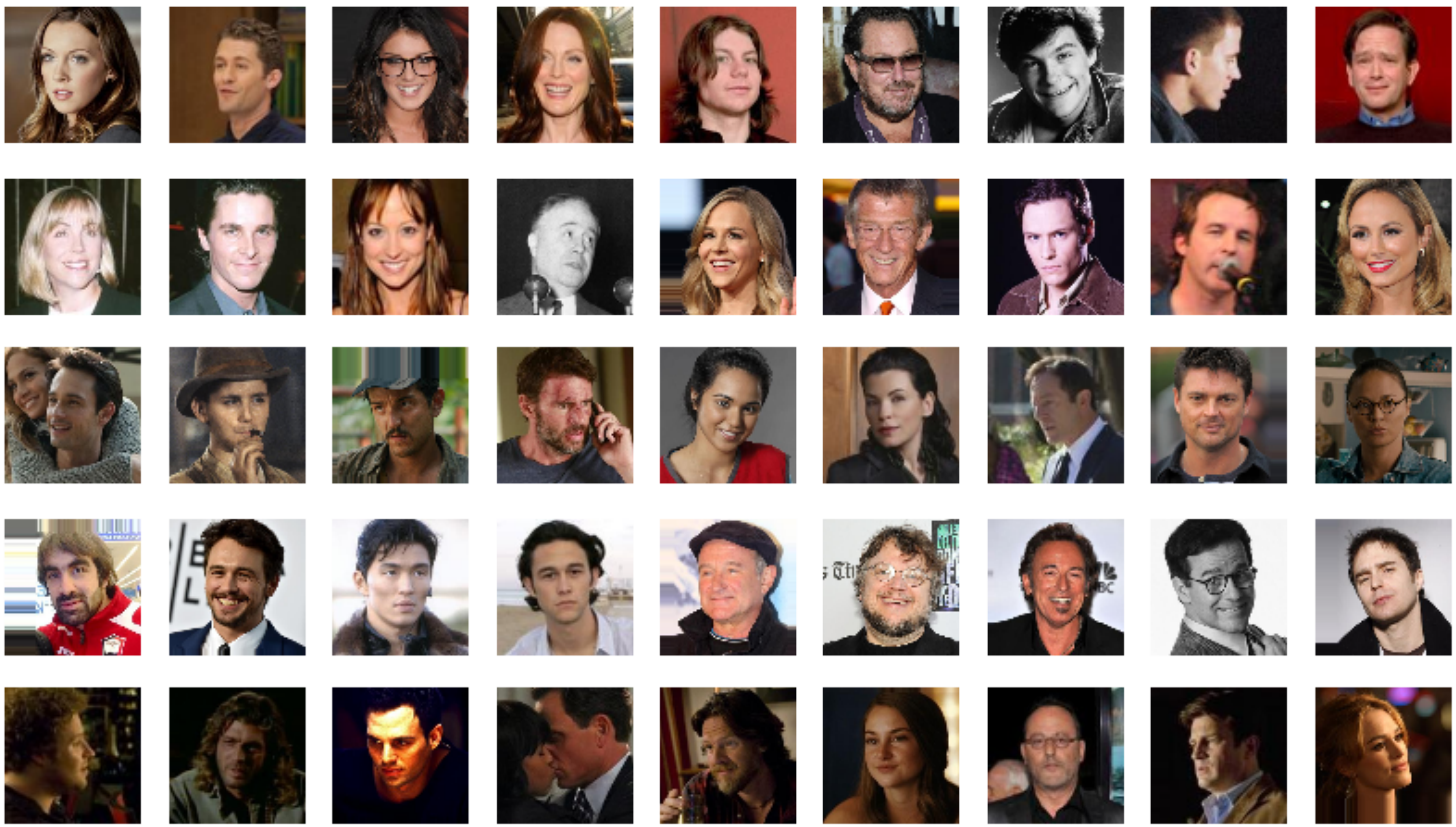
Cluster 1:

Cluster 5:

Cluster 12:

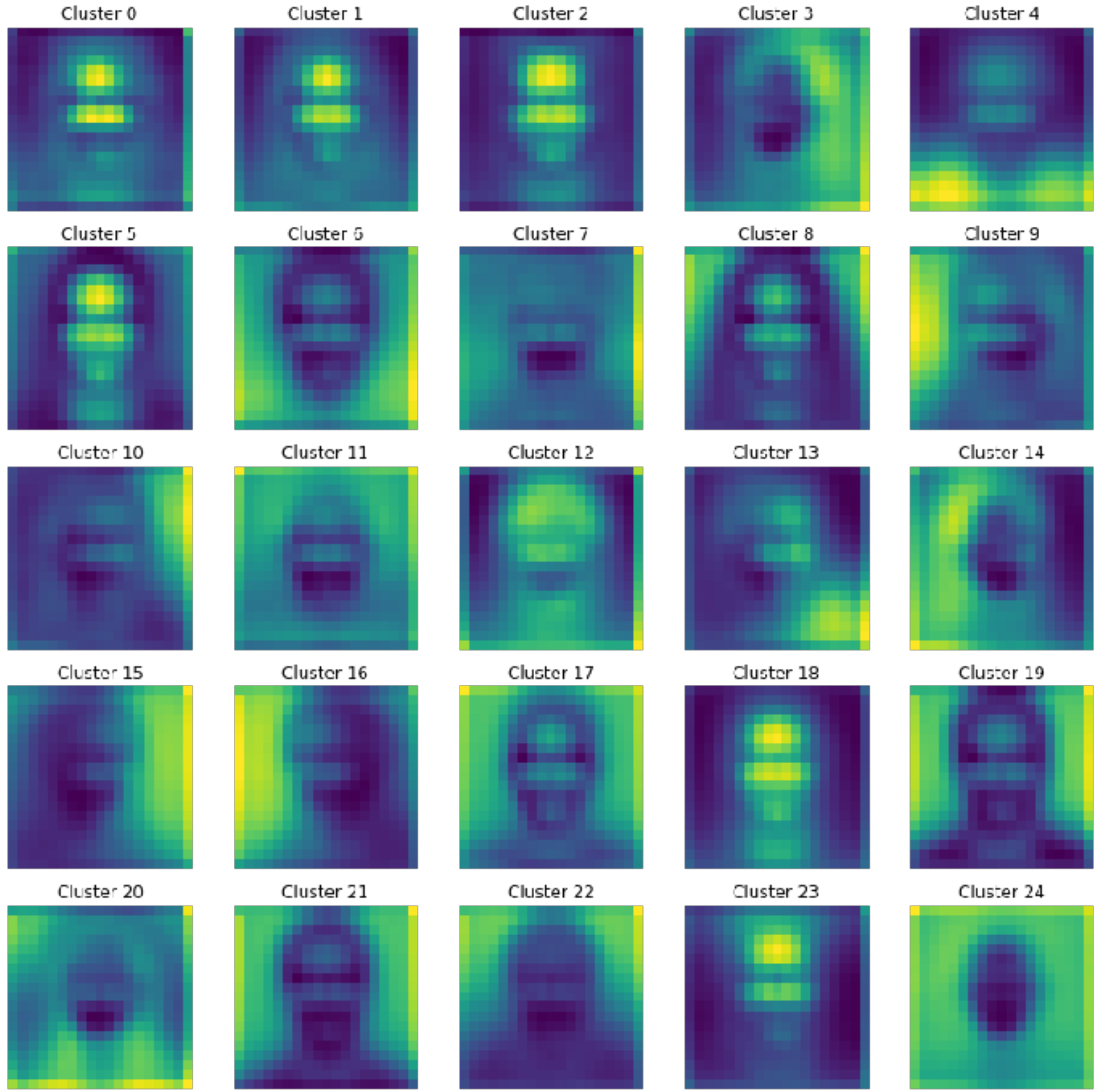
Cluster 19:

Cluster 24:



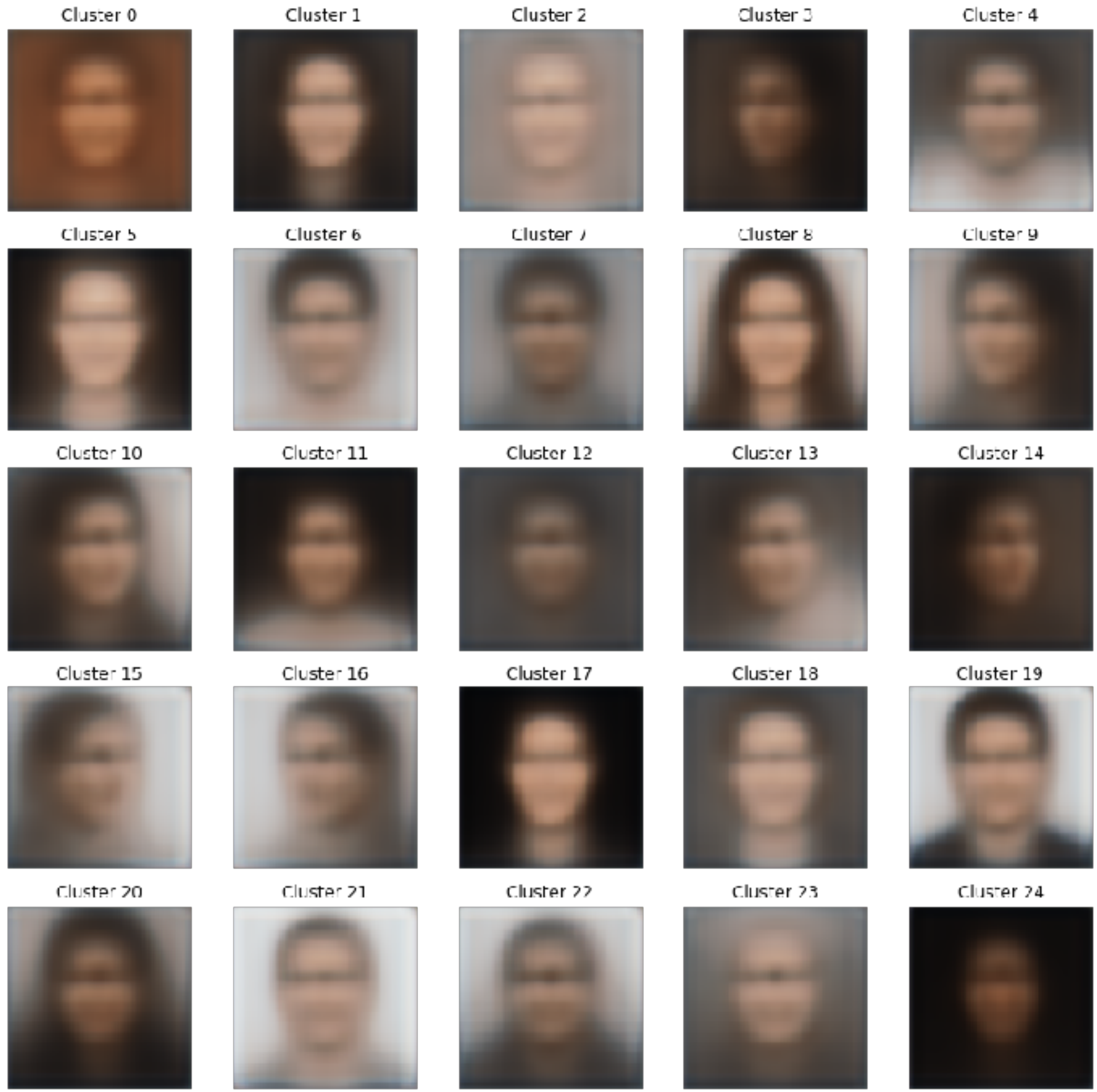
Are they any good?

- Averaged all the compressed images for each cluster.
- They are distinct!
- Some clear face shapes, hair, and background.
- What do they actually look like?



We did it!

- Decoded average compressed images
- Lighting in very important.
- This is really cool and a little scary.



Attention Map on Supervised Model

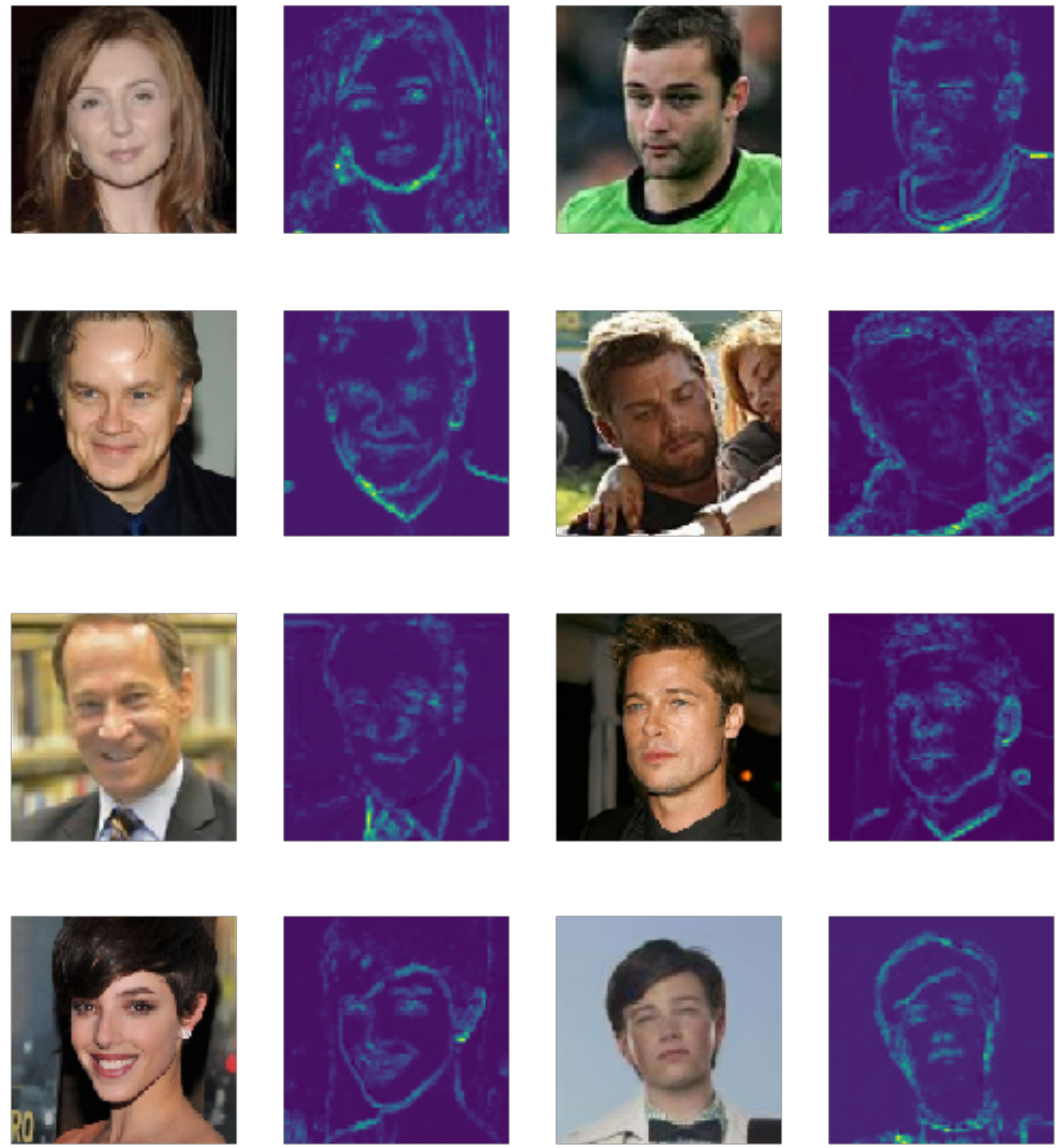
Back to our original supervised model to predict age.

```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 148, 148, 64)      1792
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 146, 146, 64)      36928
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 73, 73, 64)        0
_________________________________________________________________
dropout_6 (Dropout)          (None, 73, 73, 64)        0
_________________________________________________________________
flatten_4 (Flatten)          (None, 341056)            0
_________________________________________________________________
dense_9 (Dense)              (None, 128)               43655296
_________________________________________________________________
dropout_7 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_10 (Dense)             (None, 1)                 129
=================================================================
```

Visualize this layer!

Attention Map on Supervised Model

- Model is finding edges.
- Some edges are important for age, but it doesn't specifically look at eyes, mouths, etc.
- Overfitting is likely in the dense layers, not convolutional layers.
- We might be able to improve the model.

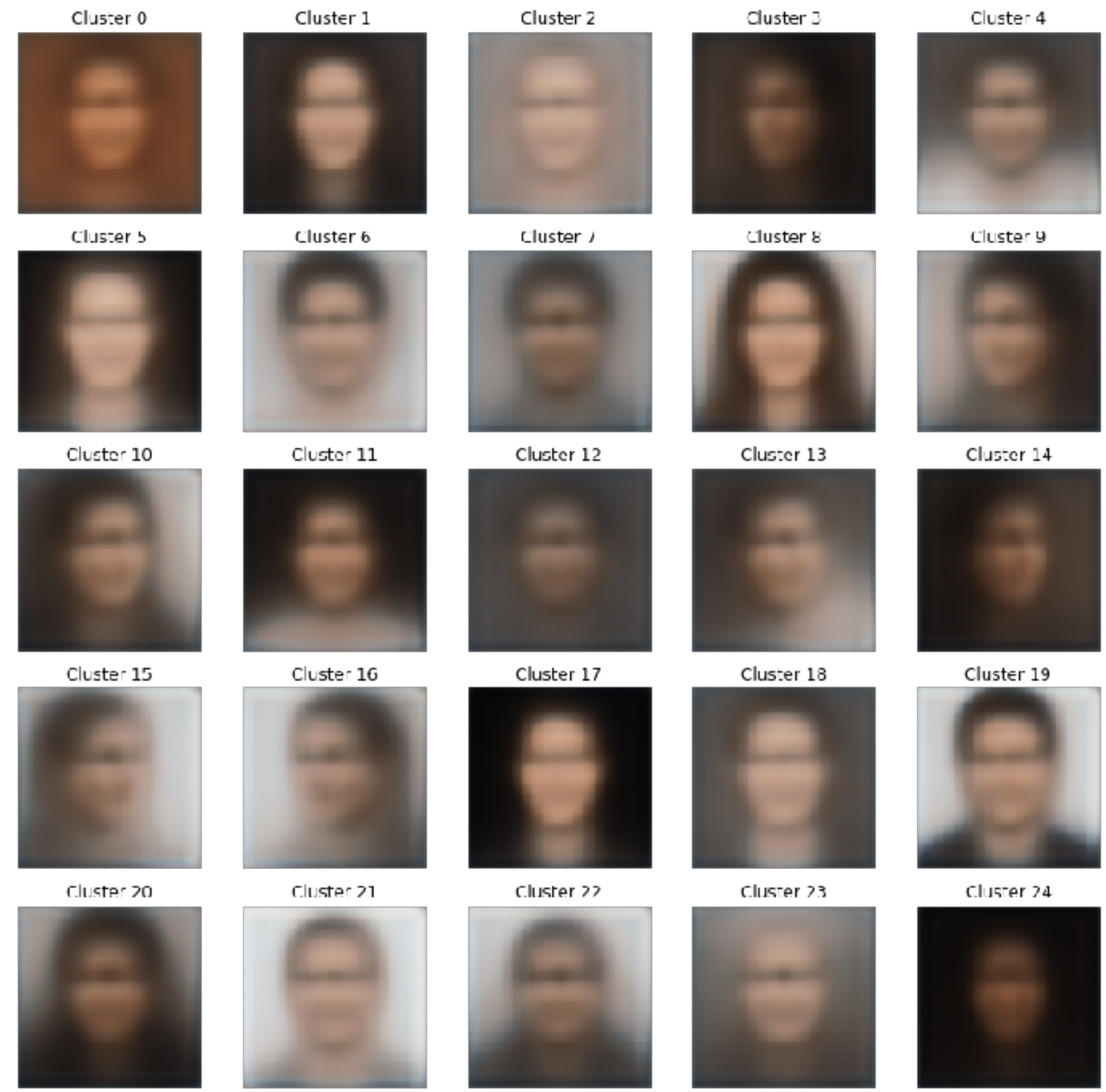


Conclusion

1. Predicting age is hard, and we did poorly.

2. Clustering based on the encoded, reduced-dimension images proved effective.

3. Attention mapping allows us to see what the convolutional layers are doing.



4. This is kind of creepy.