A MAXIMUM LIKELIHOOD ESTIMATION OF AFFINE TRANSFORMATION

The composite affine matrix M can be decomposed into individual affine matrices $M(i)$ as:

$$M = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p & 0 & 0 \\ 0 & q & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & x \\ 0 & 1 & y \\ 0 & 0 & 1 \end{bmatrix}. \tag{1}$$

By matching the elements on both sides of equation 1, we obtain:

$$\begin{cases} A_{11} = p\cos\theta + m_{11}, \\ A_{12} = q(-\sin\theta) + m_{12}, \\ A_{21} = p\sin\theta + m_{21}, \\ A_{22} = q\cos\theta + m_{22}, \\ A_{13} = px\cos\theta - qy\sin\theta + m_{13}, \\ A_{23} = px\sin\theta + qy\cos\theta + m_{23}. \end{cases} \tag{2}$$

To model the residual between the representations and the original image, we add noise $m_{ij} \sim \mathcal{N}(0, \sigma^2)$ [1]. Here we assume that the noise for each entry is independent and has the same statistics, while a more complex assumption could be made (e.g., correlated noise described by a non-diagonal covariance matrix). In order to minimize the residual, we can normalize and rearrange the equation 2:

$$\begin{cases} f_{11} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\frac{-(A_{11} - p\cos\theta)^2}{2\sigma^2}, \\ f_{12} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\frac{-(A_{12} + q\sin\theta)^2}{2\sigma^2}, \\ f_{21} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\frac{-(A_{21} - p\sin\theta)^2}{2\sigma^2}, \\ f_{22} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\frac{-(A_{22} - q\cos\theta)^2}{2\sigma^2}, \\ f_{13} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\frac{-(A_{13} - px\cos\theta + qy\sin\theta)^2}{2\sigma^2}, \\ f_{23} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\frac{-(A_{23} - px\sin\theta - qy\cos\theta)^2}{2\sigma^2}. \end{cases} \tag{3}$$

Multiplying all the f_{ij} gives us the residual likelihood function of the affine transformations:

$$f(A_{ij}, \theta, p, q, x, y) = \prod_{i=1}^{2}\prod_{j=1}^{3} f_{ij}. \tag{4}$$

The negative log-likelihood function is:

$$\begin{aligned} &-\log f(A_{ij}, \theta, p, q, x, y) \qquad (5) \\ &= 3\log(2\pi\sigma^2) + \frac{\left[\begin{aligned} (A_{11} - p\cos\theta)^2 + (A_{12} + q\sin\theta)^2 + (A_{21} - p\sin\theta)^2 + (A_{22} - q\cos\theta)^2 \\ + (A_{13} - px\cos\theta + qy\sin\theta)^2 + (A_{23} - px\sin\theta - qy\cos\theta)^2 \end{aligned}\right]}{2\sigma^2}. \end{aligned}$$

In order to find the optimal parameters to reconstruct the input θ, p, q, x and y, we take derivative against θ, p, q, x and y separately and the solution of the partial differential equations are:

---

[1] Note that the variable $m_{ij} \sim \mathcal{N}(0, \sigma^2)$ stands for zero-mean Gaussian noise, while $m \in M$ denotes the affine matrix.

$$\begin{cases} \theta & = \arctan \frac{2(A_{11}A_{21}-A_{12}A_{22})}{A_{11}^2+A_{22}^2-A_{12}^2-A_{21}^2}, \\ p & = A_{11}\cos\theta + A_{21}\sin\theta, \\ q & = -A_{12}\sin\theta + A_{21}\sin\theta, \\ x & = \frac{A_{13}\cos\theta+A_{23}\sin\theta}{p}, \\ y & = \frac{-A_{13}\sin\theta+A_{23}\cos\theta}{q}. \end{cases} \tag{6}$$

B ADIS-GAN REGULARIZER AND NON-COMMUTATIVE AFFINE MATRIX MULTIPLICATION

The affine matrix multiplication is non-commutative (unless the horizontal and vertical zoom is the same), hence different sequences of the affine transformations can produce different results. In this section, we show that the affine regularizer will not suffer from the non-commutative property by comparing the results between the two affine transformation sequences:

1. Rotation, horizontal and vertical zoom (RPQ).
2. Horizontal and vertical zoom, rotation (PQR).

B.1 ROTATION, HORIZONTAL AND VERTICAL ZOOM

$$M = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} p & 0 \\ 0 & q \end{bmatrix}. \tag{7}$$

By matching the elements on both sides of the equation 7:

$$\begin{cases} A_{11} = p\cos\theta + m_{11}, \\ A_{12} = q(-\sin\theta) + m_{12}, \\ A_{21} = p\sin\theta + m_{21}, \\ A_{22} = q\cos\theta + m_{22}, \end{cases} \tag{8}$$

where:

$$m_{ij} \sim \mathcal{N}(0, \sigma^2). \tag{9}$$

By normalizing and rearranging equation 8:

$$\begin{cases} f_{11} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \frac{-(A_{11}-p\cos\theta)^2}{2\sigma^2}, \\ f_{12} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \frac{-(A_{12}+q\sin\theta)^2}{2\sigma^2}, \\ f_{21} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \frac{-(A_{21}-p\sin\theta)^2}{2\sigma^2}, \\ f_{22} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \frac{-(A_{22}-q\cos\theta)^2}{2\sigma^2}, \end{cases} \tag{10}$$

Multiplying all the f_{ij} gives us the residual likelihood function of the affine transformations:

$$f(A_{ij}, \theta, p, q) = \prod_{i=1}^{2} \prod_{j=1}^{2} f_{ij}. \tag{11}$$

The negative log-likelihood is:

$$\begin{aligned} & -\log f(A_{ij}, \theta, p, q) \qquad (12) \\ & = 2\log(2\pi\sigma^2) + \frac{(A_{11}-p\cos\theta)^2 + (A_{12}+q\sin\theta)^2 + (A_{21}-p\sin\theta)^2 + (A_{22}-q\cos\theta)^2}{2\sigma^2}. \end{aligned}$$

In order to find the optimal parameter to reconstruct the input θ, p and q, we take derivative against θ, p and q separately and the solutions of the partial differential equations are:

$$\begin{cases} \theta_{rpq} &= \frac{1}{2}\arctan\frac{2(A_{11}A_{21}-A_{12}A_{22})}{A_{11}^2+A_{22}^2-A_{12}^2-A_{21}^2}, \\ p_{rpq} &= A_{11}\cos\theta + A_{21}\sin\theta, \\ q_{rpq} &= -A_{12}\sin\theta + A_{21}\sin\theta. \end{cases} \tag{13}$$

B.2 HORIZONTAL AND VERTICAL ZOOM, ROTATION

$$M = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} p & 0 \\ 0 & q \end{bmatrix} \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}. \tag{14}$$

By matching the elements on both sides of the equation 14:

$$\begin{cases} A_{11} = p\cos\theta + m_{11}, \\ A_{12} = p(-\sin\theta) + m_{12}, \\ A_{21} = q\sin\theta + m_{21}, \\ A_{22} = q\cos\theta + m_{22}, \end{cases} \tag{15}$$

where:

$$m_{ij} \sim \mathcal{N}(0, \sigma^2). \tag{16}$$

By normalizing and rearranging equation 15:

$$\begin{cases} f_{11} = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\frac{-(A_{11}-p\cos\theta)^2}{2\sigma^2}, \\ f_{12} = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\frac{-(A_{12}+p\sin\theta)^2}{2\sigma^2}, \\ f_{21} = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\frac{-(A_{21}-q\sin\theta)^2}{2\sigma^2}, \\ f_{22} = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\frac{-(A_{22}-q\cos\theta)^2}{2\sigma^2}, \end{cases} \tag{17}$$

Multiplying all the f_{ij} gives us the residual likelihood function of the affine transformations:

$$f(A_{ij}, \theta, p, q) = \prod_{i=1}^{2}\prod_{j=1}^{2} f_{ij}. \tag{18}$$

The negative log-likelihood is:

$$\begin{aligned} & -\log f(A_{ij}, \theta, p, q) \qquad (19) \\ &= 2\log(2\pi\sigma^2) + \frac{(A_{11}-p\cos\theta)^2 + (A_{12}+p\sin\theta)^2 + (A_{21}-q\sin\theta)^2 + (A_{22}-q\cos\theta)^2}{2\sigma^2}. \end{aligned}$$

In order to find the optimal parameter to reconstruct the input θ, p and q, we take derivative against θ, p and q separately and the solutions of the partial differential equations are:

$$\begin{cases} \theta_{pqr} &= \frac{1}{2}\arctan\frac{2(-A_{11}A_{12}+A_{21}A_{22})}{A_{11}^2+A_{22}^2-A_{12}^2-A_{21}^2}, \\ p_{pqr} &= A_{11}\cos\theta - A_{12}\sin\theta, \\ q_{pqr} &= A_{21}\sin\theta + A_{22}\cos\theta. \end{cases} \tag{20}$$

B.3 COMPARISON BETWEEN RPQ AND PQR RESULTS

If we compare the derivation results between Case 1 (rotation, horizontal and vertical zoom) and Case 2 (horizontal and vertical zoom, rotation), the expressions are different (compare equation 22 and 24). However, if we substitute the value of A_{ij} in Case 1 Equation 21 into the Equation 22 and Case 2 Equation 23 into the Equation 24 , we can find the two final expressions are the same. Hence, the affine regularizer does not change due to the non-commutative property of the affine matrix multiplication.

Case 1 (RPQ):

$$\begin{cases} A_{11} = p\cos\theta, \\ A_{12} = q(-\sin\theta), \\ A_{21} = p\sin\theta, \\ A_{22} = q\cos\theta, \end{cases} \tag{21}$$

$$\begin{cases} \theta_{rpq} & = \frac{1}{2}\arctan\frac{2(A_{11}A_{21}-A_{12}A_{22})}{A_{11}^2+A_{22}^2-A_{12}^2-A_{21}^2}, \\ p_{rpq} & = A_{11}\cos\theta + A_{21}\sin\theta, \\ q_{rpq} & = -A_{12}\sin\theta + A_{21}\sin\theta. \end{cases} \tag{22}$$

Case 2 (PQR):

$$\begin{cases} A_{11} = p\cos\theta, \\ A_{12} = p(-\sin\theta), \\ A_{21} = q\sin\theta, \\ A_{22} = q\cos\theta, \end{cases} \tag{23}$$

$$\begin{cases} \theta_{pqr} & = \frac{1}{2}\arctan\frac{2(-A_{11}A_{12}+A_{21}A_{22})}{A_{11}^2+A_{22}^2-A_{12}^2-A_{21}^2}, \\ p_{pqr} & = A_{11}\cos\theta - A_{12}\sin\theta, \\ q_{pqr} & = A_{21}\sin\theta + A_{22}\cos\theta. \end{cases} \tag{24}$$

C LOSSLESS IMAGE ENLARGEMENT ON DSPRITES

Due to the translation property of the dSprites dataset, some objects are allocated on the edge of the image frame, whereas simply enlarging the objects would cause partial information loss. Thanks to the axis alignment property of ADIS-GAN, we can enlarge the objects without information loss by firstly moving the objects to the center of the frame. The lossless image enlargement process is as follows (also see Figure 1):

1. Train an ADIS-GAN *algo*$_{\text{pxy}}$ that learns zoom, horizontal and vertical translation attributes.
2. Given an input image W_{input}, calculate the latent codes C_{input} through encoder E.
3. Calculate the affine transformation M_{input} using C_{input} through equation 1. Note that this operation is impossible without axis alignment property.
4. Calculate the inverse matrix of M_{input}, where $M_{\text{center}} = M_{\text{input}}^{-1}$.
5. Multiply the M_{center} with enlarge matrix M_{enlarge}, where $M_{\text{final}} = M_{\text{center}} \times M_{\text{enlarge}}$.
6. Use the images with enlarged objects $W_{\text{final}} = M_{\text{final}} \times W_{\text{input}}$ to train another ADIS-GAN *algo*$_{\text{r\_cat}}$ that learns rotation and shape attributes.

During the inference process (e.g., disentanglement metric calculation), the zoom, horizontal and vertical translation attributes c_{pxy} are inferenced by *algo*$_{\text{pxy}}$ and the rotation and shape attributes $c_{\text{r\_cat}}$ are inferenced by *algo*$_{\text{r\_cat}}$. The final latent representation c_{all} is inferenced by concatenating $c_{\text{r\_cat}}$ and c_{pxy}.

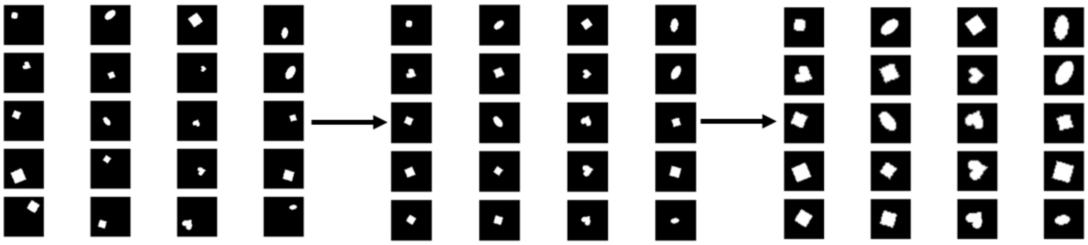


Figure 1: Illustration of objects enlargement with ADIS-GAN. a: input images. b: centered images. c: centered and enlarged images.

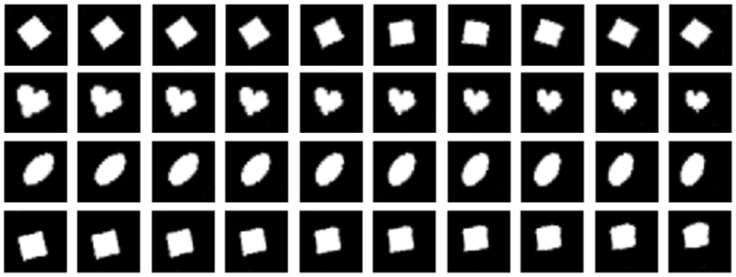


Figure 2: Disentangled representation on the dSprites dataset. The affine transformation attributes (continuous) are axis-aligned from row 1 to 4: rotation, zoom, horizontal translation, and vertical translation. The latent traversal interval is [-1,1] for continuous code. The categorical attributes from row 1 to 4: square, heart, ellipse, and square.

D FOUR SHAPES DATASET

Four shapes dataset contains 16,000 images of four shapes: square, star, circle, and triangle. It also involves the rotation attribute of the object. During the training and inference process, we resize the image from 200x200 to 28x28 for computational efficiency. The four shapes dataset can be found at:

`https://www.kaggle.com/smeschke/four-shapes`

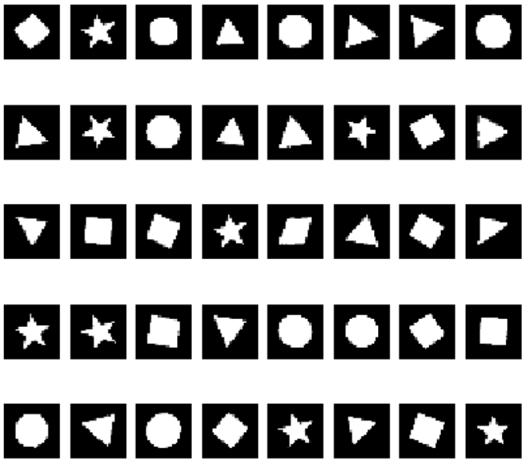


Figure 3: Samples from four shapes dataset.

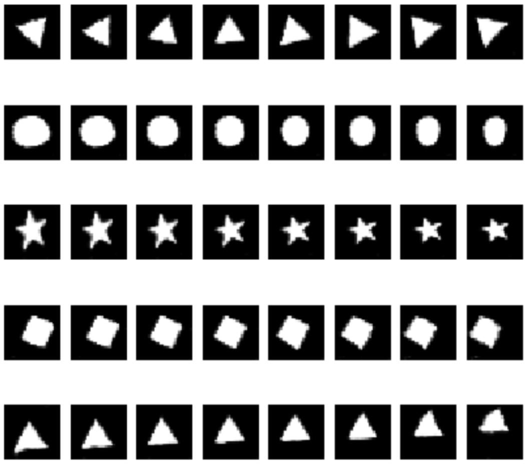
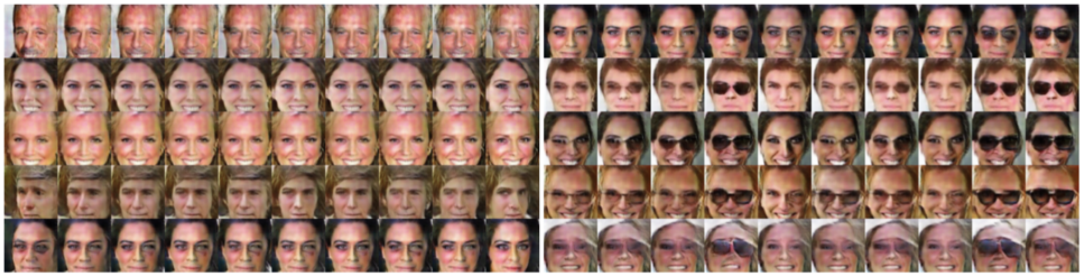
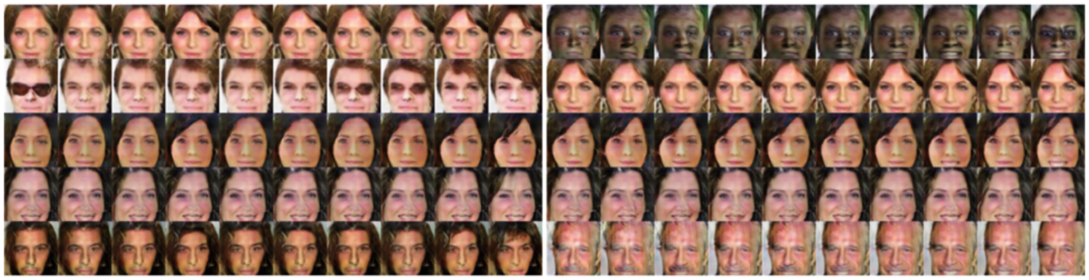


Figure 4: Disentangled representation of four shapes dataset. The continuous attributes from row 1 to 5: rotation, horizontal and vertical zoom, horizontal, and vertical translation. The categorical attributes from row 1 to 5: triangle, circle, star, square, triangle.

E CelebA Results from other algorithms

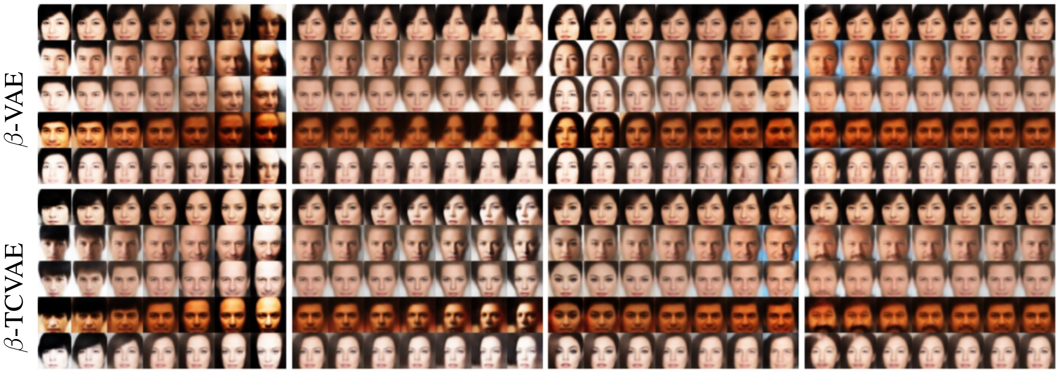


(a) Azimuth (pose) (b) Presence or absence of glasses



(c) Hair style (d) Emotion

Figure 5: CelebA results from InfoGAN Chen et al. (2016).



(a) Baldness (b) Face width (c) Gender (d) Mustache

Figure 6: CelebA results from β-VAE Higgins et al. (2017) and β-TCVAE Chen et al. (2018).

F Disentanglement Metrics

You may find the definations of the disentanglement metrics used in the experiments from Table. 1:

Table 1: Disentanglement metrics source

	FactorVAE	**DCI**	**Explicitness**	**Modularity**	**MIG**	**BetaVAE**
Paper source	Kim & Mnih (2018)	Oyallon et al. (2017)	Ridgeway & Mozer (2018)	Ridgeway & Mozer (2018)	Chen et al. (2018)	Higgins et al. (2017)

G NETWORK STRUCTURE

G.1 MNIST

The network structures are shown in Table 2. The discriminator D and the encoder E have similar networks. We use 1 ten-dimensional categorical latent code, 5 continuous latent codes and 57 noise latent codes.

Table 2: Network structure for MNIST dataset.

discriminator D/ encoder E	generator G
Input 32 × 32 gray image	Input $\in \mathbb{R}^{72}$
3 × 3 conv. 16 lReLU. stride 2. SN [2]	FC. 128 × 8 × 8 ReLU. BN
3 × 3 conv. 32 lReLU. stride 2. SN	3 × 3 upconv. 128 ReLU. stride 2. BN
3 × 3 conv. 64 lReLU. stride 2. SN	3 × 3 upconv. 64 ReLU. stride 2. BN
3 × 3 conv. 128 lReLU. stride 2. SN	3 × 3 upconv. 1 tanh. stride 1
FC. 1024 lReLU (*). SN	
From *: FC. 1. SN sigmoid for D	
From *: FC. 10. SN softmax, FC 5. SN for E	

G.2 DSPRITES

The network structures are shown in Table 3. The discriminator D and the encoder E have similar networks. We use 1 three-dimensional categorical latent code and 4 continuous latent codes for $algo_{\text{r\_cat}}$. We use 3 continuous latent codes and 69 noise latent cides for $algo_{\text{pxy}}$.

Table 3: Network structure for dSprites dataset.

discriminator D/ encoder E	generator G
Input 64 × 64 gray image	Input $\in \mathbb{R}^{72}$
4 × 4 conv. 16 lReLU. stride 2. SN	FC. 128 × 4 × 4 ReLU. BN
4 × 4 conv. 32 lReLU. stride 2. SN	3 × 3 upconv. 128 ReLU. stride 2. BN
3 × 3 conv. 64 lReLU. stride 2. SN	3 × 3 upconv. 64 ReLU. stride 2. BN
3 × 3 conv. 128 lReLU. stride 2. SN	4 × 4 upconv. 32 ReLU. stride 2. BN
FC. 1024 lRELU. SN (*)	4 × 4 upconv. 16 ReLU. stride 2. BN
From *: FC. 1. SN sigmoid for D	4 × 4 upconv. 1 tanh. stride 1
From *: FC. 3. SN softmax, FC. 4. SN for E $algo_{\text{r\_cat}}$	
From *: FC. 3. SN for E $algo_{\text{r\_pxy}}$	

---

[2] SN stands for Spectrum Normalization.

G.3 CELEBA

The network structures are shown in Table 4. The discriminator D and the encoder E have similar networks. We use 1 ten-dimensional categorical latent code, 5 continuous latent codes and 194 noise latent codes.

Table 4: Network structure for CelebA dataset.

discriminator D/ encoder E	generator G
Input 64 × 64 RGB image	Input $\in \mathbb{R}^{200}$
4 × 4 conv. 128 lReLU. stride 2. BN	4 × 4 upconv. 1024 ReLU. stride 1.
4 × 4 conv. 256 lReLU. stride 2. BN	4 × 4 upconv. 512 ReLU. stride 2. BN
4 × 4 conv. 512 lReLU. stride 2. BN	4 × 4 upconv. 256 ReLU. stride 2. BN
4 × 4 conv. 1024 lReLU. stride 2. BN	4 × 4 upconv. 128 ReLU. stride 2. BN
4 × 4 conv. 16. stride 1. (*)	4 × 4 upconv. 3 tanh. stride 2
From *: c [0:1] [3]. sigmoid for D	
From *: c [1:11]. softmax, c [11:16]. for E	

H MORE AFFINE PARAMETER COMBINATIONS

In this section, we show how to initialize the affine matrix using different combinations of affine parameters converted from latnet vectors ranging from 1-d to 7-d . Not all combinations are shown.

H.1 1-D LATENT VECTOR: ROTATION

$$M = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}. \tag{25}$$

H.2 2-D LATENT VECTOR: ROTATION, ZOOM

$$M = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} p & 0 \\ 0 & p \end{bmatrix}. \tag{26}$$

H.3 3-D LATENT VECTOR: ROTATION, HORIZONTAL AND VERTICAL ZOOM

$$M = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} p & 0 \\ 0 & q \end{bmatrix}. \tag{27}$$

H.4 4-D LATENT VECTOR: ROTATION, HORIZONTAL AND VERTICAL ZOOM, SKEW

$$M = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} p & 0 \\ 0 & q \end{bmatrix} \begin{bmatrix} 1 & m \\ m & 1 \end{bmatrix}. \tag{28}$$

---

[3] c stands for latent vector.

H.5 5-D LATENT VECTOR: ROTATION, HORIZONTAL AND VERTICAL ZOOM, HORIZONTAL AND VERTICAL SKEW

$$M = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} p & 0 \\ 0 & q \end{bmatrix} \begin{bmatrix} 1 & m \\ n & 1 \end{bmatrix}. \tag{29}$$

H.6 6-D LATENT VECTOR: ROTATION, HORIZONTAL AND VERTICAL ZOOM, HORIZONTAL AND VERTICAL SKEW, TRANSLATION

$$M = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p & 0 & 0 \\ 0 & q & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & m & 0 \\ n & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & x \\ 0 & 1 & x \\ 0 & 0 & 1 \end{bmatrix}. \tag{30}$$

H.7 7-D LATENT VECTOR: ROTATION, HORIZONTAL AND VERTICAL ZOOM, HORIZONTAL AND VERTICAL SKEW, HORIZONTAL AND VERTICAL TRANSLATION

$$M = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p & 0 & 0 \\ 0 & q & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & m & 0 \\ n & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & x \\ 0 & 1 & y \\ 0 & 0 & 1 \end{bmatrix}. \tag{31}$$

I DIFFERENT MATRIX INITIALIZATION AND MAXIMUM LIKELIHOOD ESTIMATION

In this section, we illustrate the idea of different matrix initializations and their maximum likelihood estimation with the examples of zoom and its decomposition.

I.1 HORIZONTAL AND VERTICAL ZOOM

If we initialize the affine matrix with rotation, horizontal and vertical zoom, horizontal and vertical translation:

$$M = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p & 0 & 0 \\ 0 & q & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & x \\ 0 & 1 & y \\ 0 & 0 & 1 \end{bmatrix}. \tag{32}$$

The maximum likelihood estimation is:

$$\begin{cases} \theta & = \arctan \frac{2(A_{11}A_{21} - A_{12}A_{22})}{A_{11}^2 + A_{22}^2 - A_{12}^2 - A_{21}^2}, \\ p & = A_{11}\cos\theta + A_{21}\sin\theta, \\ q & = -A_{12}\sin\theta + A_{21}\sin\theta, \\ x & = \frac{A_{13}\cos\theta + A_{23}\sin\theta}{p}, \\ y & = \frac{-A_{13}\sin\theta + A_{23}\cos\theta}{q}. \end{cases} \tag{33}$$

The latent vector is 5-d, with a less compressed and more expressive representation.

I.2 ZOOM

If we initialize the affine matrix with rotation, zoom, horizontal and vertical translation:

$$M = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p & 0 & 0 \\ 0 & p & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & x \\ 0 & 1 & y \\ 0 & 0 & 1 \end{bmatrix}. \tag{34}$$

The maximum likelihood estimation is:

$$\begin{cases} \theta & = \arctan \frac{A_{21}-A_{12}}{A_{11}+A_{22}}, \\ p & = \frac{\cos\theta(A_{11}+A_{22})+\sin\theta(A_{21}-A_{12})}{2} \\ x & = \frac{A_{13}\cos\theta+A_{23}\sin\theta}{p}, \\ y & = \frac{-A_{13}\sin\theta+A_{23}\cos\theta}{q}. \end{cases} \tag{35}$$

The latent vector is 4-d, with a more compressed and less expressive representation.

REFERENCES

Ricky T. Q. Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 31*, pp. 2610–2620. Curran Associates, Inc., 2018.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 29*, pp. 2172–2180. Curran Associates, Inc., 2016.

Irina Higgins, Loïc Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*. OpenReview.net, 2017.

Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2649–2658, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.

E. Oyallon, E. Belilovsky, and S. Zagoruyko. Scaling the scattering transform: Deep hybrid networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5619–5628, 2017.

Karl Ridgeway and Michael C Mozer. Learning deep disentangled embeddings with the f-statistic loss. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 31*, pp. 185–194. Curran Associates, Inc., 2018.