# Unsupervised Attribute Alignment for Characterizing Distribution Shift

**Anonymous Author(s)**
**Affiliation**
**Address**
email

## Abstract

Detecting and addressing distribution shift is an important task in machine learning. However, most of the machine learning solutions to deal with distribution shift lack the capability to identify the key characteristics of such a shift and present it to humans in an interpretable way. In this work, we propose a novel framework to compare two datasets and identify distribution shifts between the datasets. The key challenge is to identify generative factors of variation, which we refer to as *attributes*, that characterize the similarities and differences between the datasets. Producing this characterization requires finding a set of attributes that can be aligned between the two datasets and sets that are unique. We address this challenge through a novel approach that performs both attribute discovery and attribute alignment across the two distributions. We evaluate our algorithm's effectiveness at accurately identifying these attributes in two separate experiments, one involving two variants of MNIST and a second experiment involving two versions of dSprites.

## 1 Introduction

Machine learning (ML) algorithms traditionally assume that data is drawn from a single, stationary distribution. In many real world scenarios, however, such an assumption likely will not hold, leading to uncontrollable or sometimes catastrophic failures [21, 8]. As a result, detecting, quantifying, and compensating for the changes in distribution are of paramount importance. Many recent methods propose making models more robust to arbitrary shifts in the data at test time [15, 17, 8, 19]. Most of these techniques, as well as the efforts to make models more robust, have largely focused on learning to make invariant predictions despite these shifts, treating them as noise or nuisance variables.

In many cases, characterizing precisely how the test distribution is different from the training distribution can be extremely useful. Unfortunately, despite the large body of work in this space, surprisingly few studies focus on characterizing shifts and communicating this information in a form that is understandable to human users [18].

Characterizing distribution shift is especially important in scientific applications where ML models are first trained in a simulator [4, 14] before being applied to make inferences on real-world data. Simulators provide a cost-effective way to generate more samples in the experimental design space, but the fidelity of the simulated data to real-world data is an obvious concern. Characterizing the distribution shifts between simulated and real-world data can give us insights into the knowledge gap in our current understanding of scientific principles that need to be addressed.

One critical component to achieve such a system is the unsupervised discovery of the underlying generative factors, or attributes, in a data distribution. With the development of high quality generative models [6, 13, 11], this goal can be achieved using unsupervised attribute discovery methods [7, 20, 22]. Here, concept vectors [12], i.e. linear directions in the latent space of a generative model, that correspond to specific data attributes, are identified; moving along these directions induces meaningful changes to the semantics of an output image. However, to characterize the difference

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

between two distributions, it is crucial to align them first. Generative methods have been used extensively to map between two distributions, implicitly achieving alignment between them, like unpaired image translation methods using CycleGAN [23] and one-to-many mapping by separating image content and style using MUNIT [10]. Most of these methods [2] are motivated by, and geared toward, generating high quality translated images rather than explaining shifts. In other words, they help identify "parallel" or analogous attributes in both the distributions, implicitly making an assumption that a one-to-one mapping exists between the two datasets, which is highly restrictive. The problem of interest here is identifying similar and dissimilar attributes given two pre-trained generative models, particularly when the assumption of a one-to-one mapping cannot be made.

In this work, we aim to fill in this gap and introduce a novel approach to provide attribute level characterization of the distribution shift. We achieve this through a joint framework that combines attribute discovery and attribute alignment across distributions. Our method identifies linear directions that encode both unique and shared attributes from the respective generative models derived from the original data distributions. Our key contributions are as follows: 1) to propose a novel perspective to characterize the distribution shift through attribute alignment across datasets and 2) to introduce a framework that solves the joint attribute discovery and attribute alignment problem.

## 2 Methodology

Unsupervised attribute alignment between data distributions requires us to solve both the attribution discovery as well as the alignment tasks. Ideally, we want to solve both tasks simultaneously so that attributes can be more reliably identified. Many existing unsupervised GAN attribute discovery works [7, 20] rely on closed-form optimization to extract attribute directions, thereby requiring a post-hoc attribute alignment step. In comparison, the recent work by Voynov *et al.* [22] approaches the problem from a rather different perspective, in which a predictive model can be learned together with the concept directions for predicting the respective semantic shifts. In this work, we leverage the flexibility of a similar formulation to solve the joint optimization problem.

**Formulation** Let $G$ be a pre-trained generator network that learns a mapping from a prior distribution, e.g., Gaussian $P(z) = \mathcal{N}(0, I^d)$, to a data distribution $G : P(z) \mapsto P(X)$. Here, we define an attribute as a direction, $w \in R^d$, in the latent space, that corresponds to a meaningful semantic change as $G(z) \rightarrow G(z + \epsilon w)$, where the extent of change is determined by $\epsilon$. As such, attribute discovery is the problem of identifying such directions to recover known semantic factors (labels such as digit thickness, shape, color etc.) in a data distribution. To make this explicit, we associate the $k^{th}$ direction, $w_k$, with a known semantic factor, $\alpha_k \approx w_k$.

Next, we describe the unsupervised attribute discovery problem for a single generative model formulated in [7], which we extend to multiple models subsequently. Let $W^{K \times d}$ be a learnable matrix, with hyper-parameter $K$ corresponding to the total number of attributes to discover from the data distribution, applied to the $d$-dimensional latent space of the generative network. The $W$ matrix contains all the possible directions $W = [w_1, \ldots, w_K]$, used to explore the latent space. In order to identify meaningful changes in an unsupervised fashion, $W$ is trained based on the fact that these directions are expected to be predictable – i.e., a discriminator network, $D$, is trained to take a pair of generated data as input, $(G(z), G(z + W(\epsilon k))$, to predict both $\epsilon$ and the specific direction $k$ that induced the change.

The output of $D$ is a probability distribution over the $K$ attributes to learn $P_K$ and a prediction of the scalar $\hat{\epsilon}$. Models $D$ and $W$ are jointly trained to minimize the cross entropy loss $L_{CE}$ of $D$'s prediction of attribute $k$, where $k$ is sampled from a uniform distribution $k\ U\{1, K\}$, and the mean absolute error Loss $L_{MAE}$ between $\epsilon$ and $\hat{\epsilon}$.

$$L(W, D) = L_{CE}(D(G(z), G(z + W(\epsilon k)), k) + \lambda * L_{MAE}(\epsilon, \hat{\epsilon})$$

Where $\lambda = .25$ is a hyperparameter to weight the $L_{MAE}$

**Attribute discovery for multiple generators** We now extend this setup to the case of multiple generative models, in order to characterize distribution shifts using attribute level alignment. Let $G_1$ and $G_2$ be two pre-trained generators for both the data distributions of interest, and $W_1^{K_1 \times d_1}$ and $W_2^{K_2 \times d_2}$ are the learnable matrices. Unlike the case for a single model, the discriminator network $D$, must act on two sets of image pairs (one normal-perturbed pair from each generator). Moreover, we

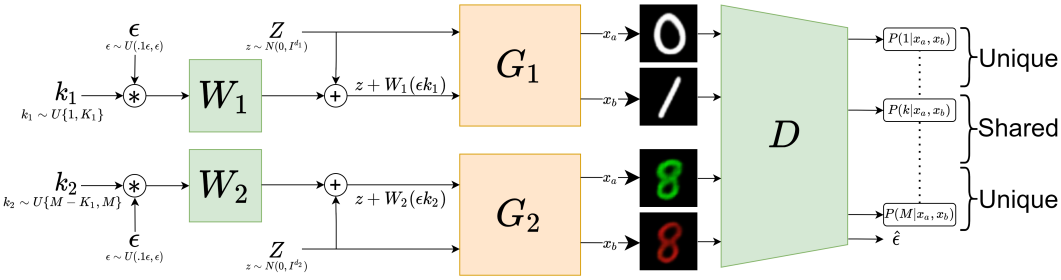


Figure 1: An illustration of the proposed approach.

modify its output to be $M$ different attributes, such that they are composed of $K_1$ attributes from the first data distribution and the $K_2$ attributes of the second data distribution. $M$ is composed of unique attributes for both data distributions, as well as shared directions $M > K_1$ and $M > K_2$. If dataset one is a strict subset of dataset two, then we can define $M = K_2$ and $K_1 < K_2$.

Since this process requires estimation of the attributes and alignment of them to be similar across a distribution shift, the problem is fairly ill-posed with a high likelihood of reducing to trivial solutions. As such, we place different kinds of constraints (orthogonality, sparsity, disentanglement) to prevent these failure cases. We illustrate the entire model in Figure 1.

**Generative model choice** Since the joint alignment and discovery problem is ill-posed, we find that disentanglement of the latent space, using a $\beta$-VAE [9] helps improve the optimization to find good solutions. As a result, in our studies we use the beta-VAE's decoder as our generative model $G$.

However, disentanglement on its own is insufficient, since there is a risk of the model combining multiple attributes that may *appear* unique to $D$, resulting in a failure to find the truly unique attributes. As a result, we apply both a sparsity regularization and an additional spectral norm orthogonality loss [1] on the learned $W_1$ and $W_2$ matrices. The sparsity term ensures that not too many independent attributes are being modified at the same time (e.g. preventing a direction that modifies thickness, color, and digit as the same time). Orthogonality loss ensures any two learned directions are not nearly identical. For example, without this loss, many directions would learn the same transformation into an eights digit, but with slightly altered rotations. The sparsity loss is enforced using a simple $\ell_1$-norm constraint:$L_{sparse}(W) = \sum_{i,j} ||W_{i,j}||_1$, and the orthonormality constraint : $L_{ortho}(W) = \sigma(W^T W - I)$, where $\sigma$ is the spectral norm. Finally, the overall cost function is denoted as follows:

$$L_{\text{total}} = L_{ce} + \lambda_\epsilon L_{MAE} + \lambda_s(L_{sparse}(W_1) + L_{sparse}(W_2)) + \lambda_o(L_{ortho}(W_1) + L_{ortho}(W_2)) \quad (1)$$

for all experiments we set $\lambda_\epsilon = 0.25, \lambda_s = 0.1,$ and$\lambda_o = 0.01$

**Unidirectional Attributes** In the original formulation of unsupervised attribute discovery, the scalar value $\epsilon$ is sampled uniformly from $U(-\epsilon, \epsilon)$. This seems reasonable as a learned attribute can be added or removed (e.g. thick digits in the positive direction versus thin digits in the negative direction). However, having negative $\epsilon$ values does not work when aligning two data distributions. While some semantic attributes, such as thickness, make sense to be linearly defined in both the positive and negative directions, there is too much flexibility in the learned latent space. A positive direction may land in a region corresponding to a certain digit, where the negative direction could be any other arbitrary digit. In order to align two datasets, both would have to have learn bidirectional attributes that correspond to identical class pairs. For this reason, we simply change the scalar to be sampled uniformly from $U(0.1 * \epsilon, \epsilon)$.

## 3 Experiments

**Datasets** As a first foray into attribute alignment, we selected two experiments with attributes that can be semantically identified quickly. To achieve this effort, we design two experiments where one data distribution is a strict subset of the other: one distribution will contain only aligned attributes, and the other will contain both aligned and unique attributes. Therefore, we conduct

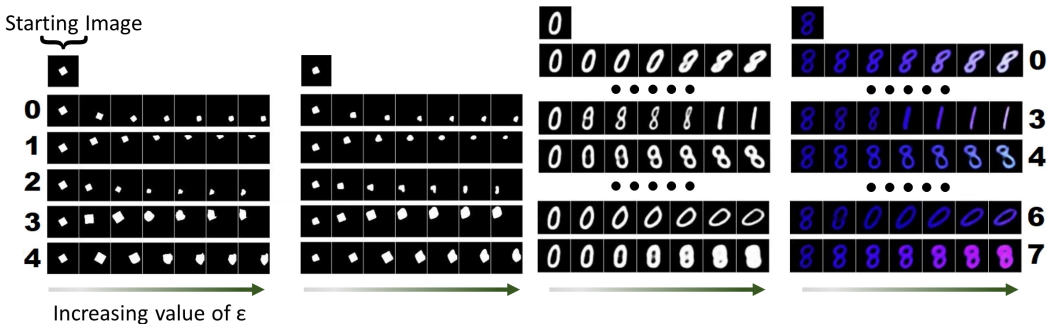


Figure 2: Results from our experiments demonstrating the shared learned attributes. **(Left):** dSprites squares versus full dSprites, where various object translations are the learned changes. **(Right):** select example attributes from MNIST vs colored MNIST, where the learned attributes are class, orientation and intensity. Some attributes are non-intuitive: Colored MNIST row 7, despite looking like an erroneous color change, makes sense as the overall intensity of the RGB values is increasing.

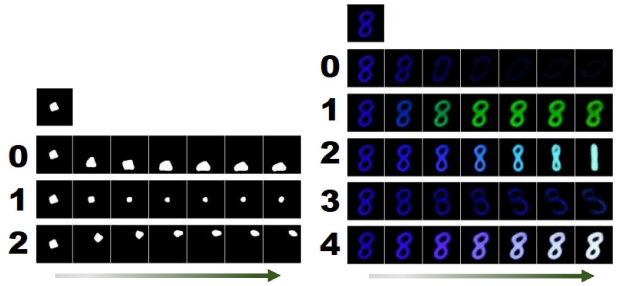


Figure 3: The unique directions. dSprites correctly learned hearts (row 1), and ellipses (rows 2 and 3). Colored MNIST is fair, learning the unique color related attributes in row 1 and 2.

two main experiments: MNIST [3] versus colored-MNIST [5], and dSprites versus square dSprites [16]. MNIST contains images with a single channel and 28x28 dimensions, where colored MNIST randomly applies color to a given image to get 3-channeled instances. dSprites is a synthetic dataset of 5 independent attributes (location, scale, rotation, shape), and we create a modified version with only square shapes. Our goal is to leverage the learnings from these datasets to then apply them to more sophisticated distributions, such as faces versus anime faces, or experimental scientific data versus simulated scientific data.

**Results** We show the results of attribute alignment for our two experiments in Figures 2 and 3. The former shows the learned, aligned, attributes between (left) dSprites squares and full dSprites and (right) MNIST and colored MNIST. The dSprites experiment mostly learns positional alignment, moving a given shape between the cardinal positions in directions 0 to 3, and rotation in direction 4. The MNIST experiment shows changes in the digit class, the orientation of a digit, and the intensity of the instance. We present more examples for both experiments, along with implementation details, in the supplemental material.

## 4 Discussion

The proposed method provides the first unsupervised attribute alignment method (to the extent of our knowledge) for characterizing distribution shift. We solve the problem through a joint optimization of attribute discovery and attribute alignment. It provides an intuition and an accessible way to characterize semantics shifts between distributions. As the first solution to this particularly challenging problem, we are aware and plan to improve upon certain limitations of the proposed approach. Since we learn the alignment regarding to the latent spaces of generative models, the quality and property of these models would have a great impact on the final quality of the result. Moreover, the current approach makes the assumption that semantically meaningful attributes are more predictive for the attribute discovery [22], which may not hold for true in a more general setting. To this end, a more effective attribute search scheme/criteria is likely required to make the overall system robust and applicable to wider range of generative models.

## References


[1] Nitin Bansal, Xiaohan Chen, and Zhangyang Wang. Can we gain more from orthogonality regularizations in training deep cnns? *arXiv preprint arXiv:1810.09102*, 2018.

[2] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8188–8197, 2020.

[3] Li Deng. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[4] Francesco Di Natale, Harsh Bhatia, Timothy S Carpenter, Chris Neale, Sara Kokkila-Schumacher, Tomas Oppelstrup, Liam Stanton, Xiaohua Zhang, Shiv Sundram, Thomas RW Scogland, et al. A massively parallel infrastructure for adaptive multiscale simulations: modeling ras initiation pathway for cancer. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–16, 2019.

[5] Abel Gonzalez-Garcia, Joost van de Weijer, and Yoshua Bengio. Image-to-image translation for cross-domain disentanglement. 2018.

[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[7] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. GANspace: Discovering interpretable GAN controls. *arXiv preprint arXiv:2004.02546*, 2020.

[8] Dan Hendrycks and Thomas G. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL `https://openreview.net/forum?id=HJz6tiCqYm`.

[9] Irina Higgins, Loic 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. 2016.

[10] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *ECCV*, 2018.

[11] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020.

[12] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pages 2668–2677. PMLR, 2018.

[13] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[14] Bogdan Kustowski, Jim A Gaffney, Brian K Spears, Gemma J Anderson, Jayaraman J Thiagarajan, and Rushil Anirudh. Transfer learning as a tool for reducing simulation bias: application to inertial confinement fusion. *IEEE Transactions on Plasma Science*, 48(1):46–53, 2019.

[15] Himabindu Lakkaraju, Ece Kamar, Rich Caruana, and Eric Horvitz. Identifying unknown unknowns in the open world: Representations and policies for guided exploration. In *Proceedings of the AAAI Conference on Artificial Intelligence*, page 2124–2132, 2017.

[16] Loic Matthey, Irina Higgins, Demis Hassabis, and Alexander Lerchner. dsprites: Disentanglement testing sprites dataset. https://github.com/deepmind/dsprites-dataset/, 2017.

[17] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, pages 69–84. Springer, 2016.

[18] Matthew L. Olson, Thuy-Vy Nguyen, Gaurav Dixit, Neale Ratzlaff, Weng-Keen Wong, and Minsuk Kahng. Contrastive identification of covariate shift in image data. In *2021 IEEE Visualization Conference (VIS)*. IEEE, 2021.

[19] Fengchun Qiao, Long Zhao, and Xi Peng. Learning to learn single domain generalization. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 12553–12562. IEEE, 2020. doi: 10.1109/CVPR42600.2020.01257. URL `https://doi.org/10.1109/CVPR42600.2020.01257`.

[20] Yujun Shen and Bolei Zhou. Closed-form factorization of latent semantics in GANs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1532–1540, 2021.

[21] Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.

[22] Andrey Voynov and Artem Babenko. Unsupervised discovery of interpretable directions in the gan latent space. In *International Conference on Machine Learning*, pages 9786–9796. PMLR, 2020.

[23] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

# A Appendix

**Implementation details** We train a $\beta$-VAE on all 4 datasets. We use a latent size $d = 10$ for mnist experiments and $d = 5$ for dsprite experiments. The values of $K_1$ and $K_2$ are somewhat arbitrary, so we set them to the value of the latent size for the subsets, set d-sprites $K_2 = 6$ and set colorminst is set to $K_2 = 15$.

We use a LeNet backbone as our discrimintive model $D$ for all our experiments. The latent points $z$ are always sampled from a normal distribution $N(0, I)$ matching the dimensionality of the original generator. The direction index $i$ is sampled from the uniform distribution $U\{1, K_1\}$ or $U\{1, K_2\}$. We find using a shift scalar multiplier $\epsilon$ maximum of 3 for both all experiments to provide the most consistent training results, however this value is not always large enough at test time.

In order to ensure the $l_2$-norms of our learned $W$ matrices are not too high, we force all $W$ matrix columns to have a unit vector length, as the cross entropy loss learns $W$ with high norms to simplify classification without the constraint.

Lastly, all model are jointly optimized using an Adam optimizer with learning rate of $0.01$, trained for 5000 iterations with a mini-batch size of 32. We found additional training for these relatively lower dimensional image datasets to be unnecessary.

**Additional examples** Figure 4 shows all learned directions from the main text's example in Figure 2. Furthermore, we provide additional random samples from our experiments. Figure 5 and 6 shows a side-by-side example of the shared directions for our MNIST and dSprites experiments respectively; figures 7 and 8 show the unique attributes learned for MNIST and dSprites.

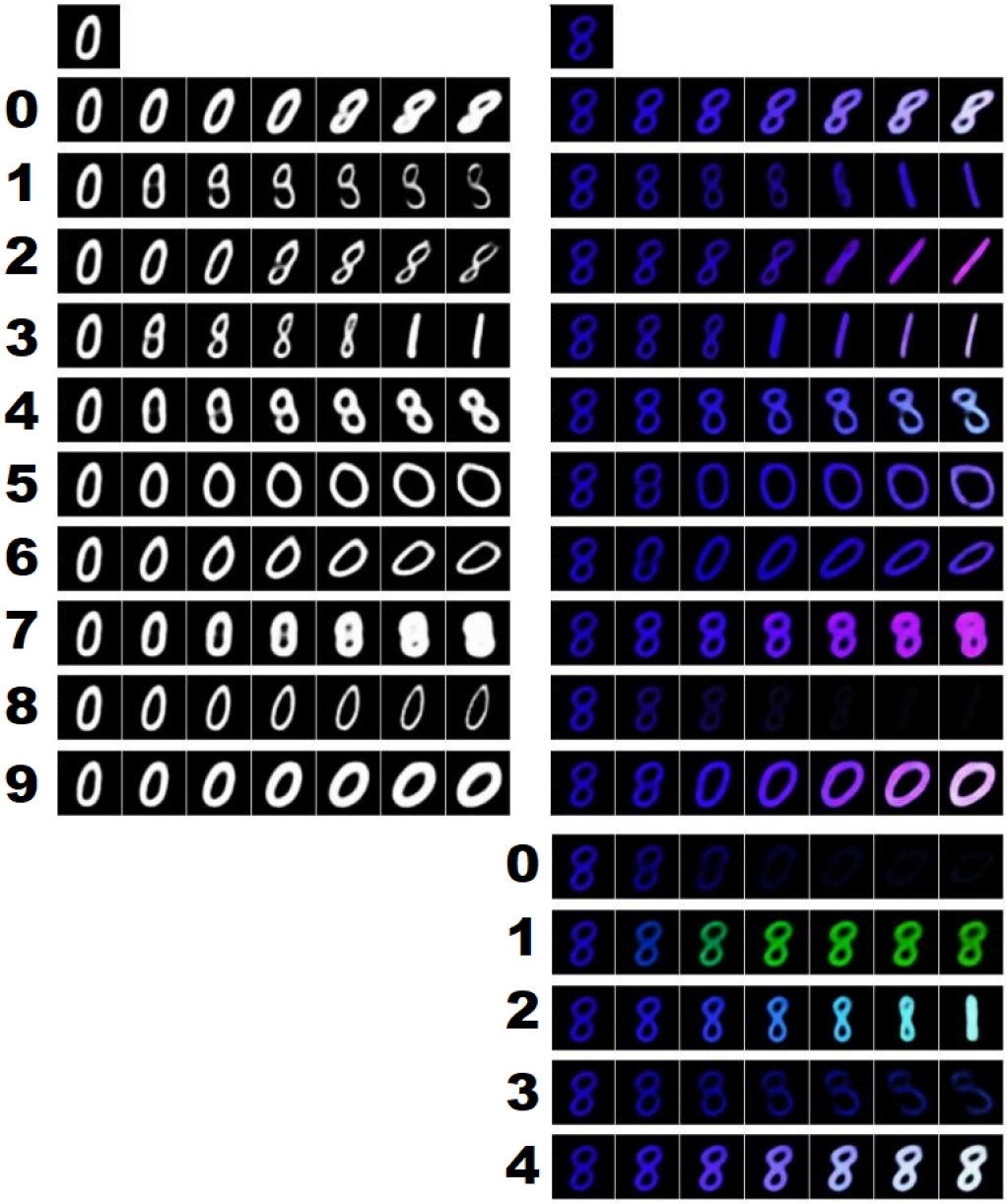


Figure 4: All learned attributes for MNIST vs colored MNIST.

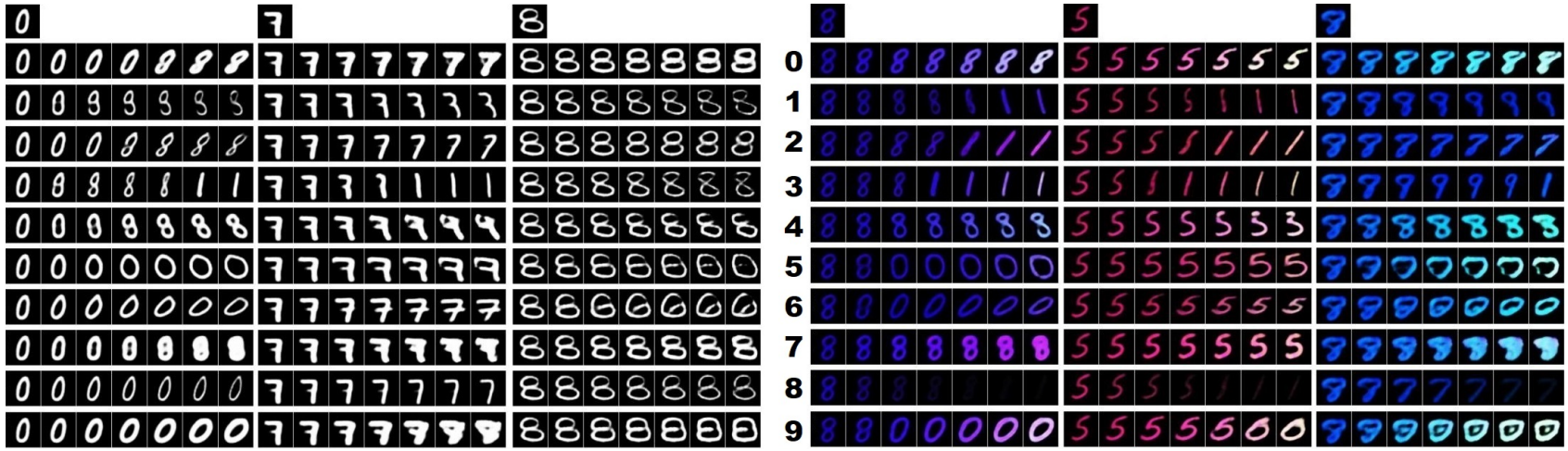


Figure 5: mnist vs cmnist shared directions.

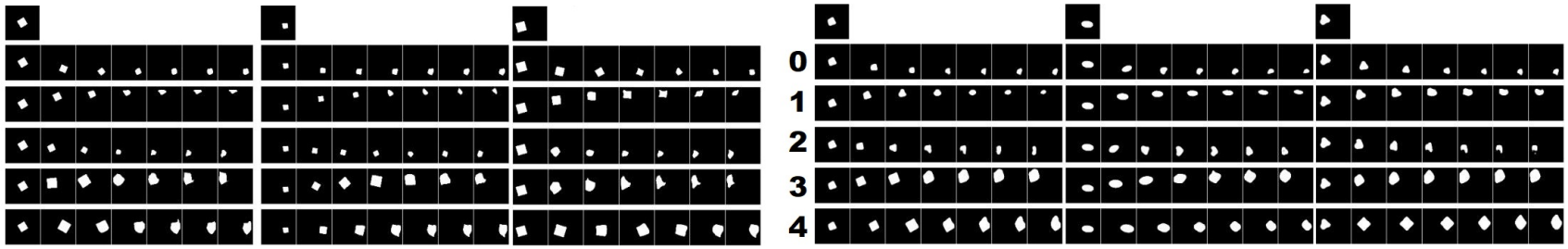


Figure 6: squares vs full dsprites shared directions.

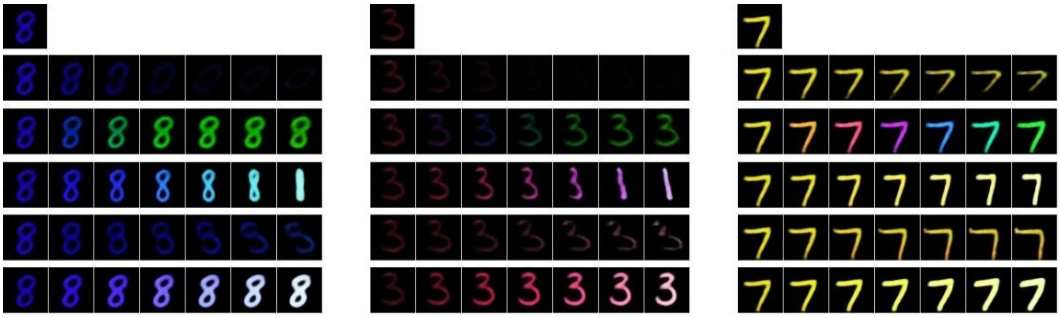

Figure 7: cmnist unique directions.

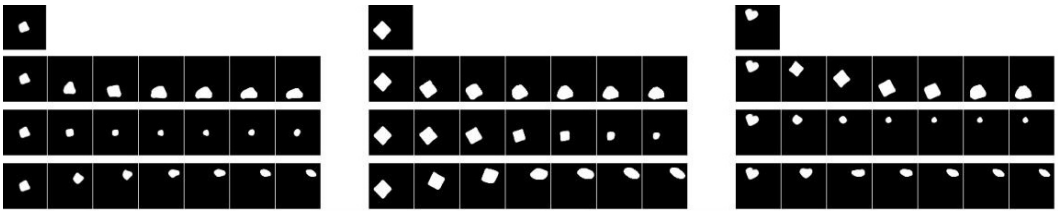

Figure 8: dSprites unique directions.