Supervising Variational Autoencoder Latent Representations with Language

Thomas Lu, Aboli Marathe, Ada Martin *
Machine Learning Department
Carnegie Mellon University
Pittsburgh, PA 15213
ttlu, abolim, gamartin@cs.cmu.edu

Abstract

Supervising latent representations of data is of great interest to modern multi-modal generative machine learning. In this work, we propose two new, simple methods to use text to condition the latent representations of a VAE and evaluate them on a novel conditional image-generation benchmark task. We find that the applied methods can be used to generate highly accurate reconstructed images through language querying. Our methods are quantitatively successful at conforming to textually-supervised attributes of an image while keeping unsupervised attributes constant. At large, we present critical observations on disentanglement between supervised and unsupervised properties of images and identify common barriers to effective disentanglement.

1 Introduction and Motivation

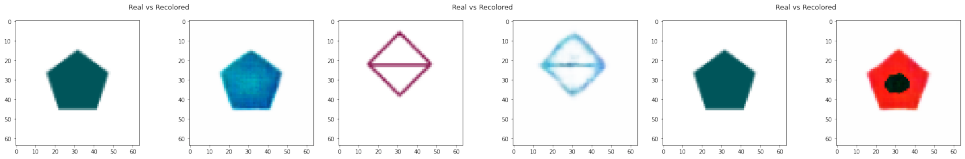


Figure 1: Feature VAE with arbitrary color inputs. The first two on the left were recolored to the prompt "Dolphin", **which is not in our dataset**. The last was recolored to the prompt "Carnegie Mellon", which has red school colors, also not in our dataset.

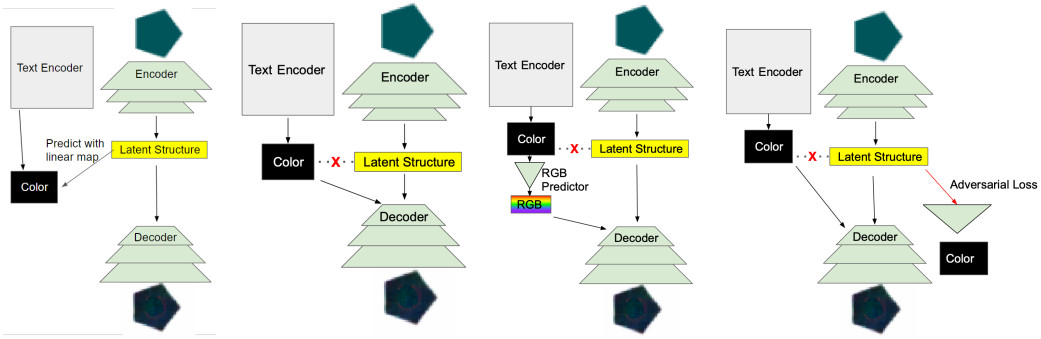
VAEs are a popular modern recipe for the encoding of complex data. However, the latent features discovered by a standard VAE are not directly controllable nor interpretable. The notion of disentangling the latent space involves separating recognizable features such that different portions of the latent vector contain independent generative parameters. This has applications such as debiasing, bias analysis, and many generation subtasks. Van et al.[28] proposes a method for supervising the latent space of generative adversarial networks to ensure they correlate to some known feature of the input. As a result, they can modify the latent representation of an input image along one of these supervised dimensions to create an alternate version of that image with that property modified. In this report, we extend this supervision strategy to a VAE model over textual descriptions of features.

Similarly, we provide a second method to condition VAE models by taking advantage of the information-theoretic constraints arising from the objective function. This method only utilizes simple vector concatenation as the main mechanism for conditioning.

*Equal Contribution

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

We explore the extension of these strategies to language supervision on an algorithmically-generated colored shape dataset. In particular, we evaluate on the task of image recoloring. The model receives an image and a natural language target color as input. It must then output the image recolored to the target color. We find that both of our methods can separate labeled feature information from unlabeled feature information within the latent space, and can be used to generate highly accurate recolored images by directly inputting a language query at evaluation time. While other works have tackled this same task, we hope that this work can serve as an initial inquiry into new directions of performing text-conditioned image generation. The code for this work is released in our GitHub repository[2]. Our models were designed for ease of implementation and experimentation without extensive computational resources. All code was run using a 2018 MacBook Air.



(a) SLt-VAE (b) Feature VAE (c) Weakly supervised Feature VAE (d) Feature VAE w. Adversary

Figure 2: Diagrams for proposed models. Trained parameters are in green.

The main contributions we hope to highlight in this work are as follows:

1. We introduce two simple methods for text-based conditional image generation using VAEs. The first, dubbed SLt-VAE, linearly predicts text embeddings from the latent space, whereupon a simple linear algebraic method allows conditional modification of the input image.
2. The second method, dubbed Feature VAE, concatenates the text embedding features to the latent representation prior to decoding. Due to information theoretic principles, the typical VAE loss function naturally encourages the model to separate any information encoded by the text embedding and information encoded in the latent vector. Replacement of the concatenated vector with the latent vector corresponding to an alternate textual description allows arbitrary recoloring of the image.
3. We provide analysis and preliminary experiments to improve these methods and further our understanding of VAE disentanglement.
4. We provide a simple dataset for simple disentanglement tests. This dataset sports a small download size, many labeled auxiliary features, and a low-resolution size for low-compute experimentation.

2 Related Work

Variational auto-encoders [15] were introduced in 2013 as an efficient learning algorithm for modeling continuous latent variables. These networks were quickly adopted for tasks spanning generation [26], anomaly detection [2], and unsupervised learning paradigms [6].

Newer models attempted to build on the VAE through specific learning mechanisms [27, 31] attempting to exploit the naturally regularized latent space for solving more complex tasks. A recent work [29] improved performance on sentiment analysis by training a classifier on the intermediate representation, which was divided between known properties and latent variables, which is a method bearing some similarities to our baselines.

[2]https://github.com/thomaslu2000/vae-latent-text-alignment

Several works have delved into understanding auto-encoders for robust learning of disentangled representations[3]. One such VAE introduced in 2017 was β-VAE [11] which presented a method to learn interpretable factorized latent representations in a completely unsupervised manner by increasing the KL divergence loss term to control the information of the latent representation, encouraging features to avoid redundant information. Many works have furthered the concept of disentanglement and conditional image manipulation, such as StyleGANs [14], CCVAEs [13], and various adversarial methods [18] [10].

Our work on the SLt-VAE is most immediately informed by Van et al.[28], which adds an additional loss to a GAN encouraging its latent space to align with some supervised information about a given sample. SLt-VAE adapts this method to text through the use of pretrained embedding models such as CLIP [23], as well as to VAEs, allowing them to operate over any input image. We also rework the image manipulation formula and remove the orthogonality constraint.

Our F-VAE bears some similarities to LORD [9], which also separates labeled class and content information, but combines both in the same generator. Other works, such as ZeroDIM, have also found positive results by implementing CLIP and its powerful generalization abilities [8]. However, our model simply uses the CLIP embedding directly rather than training additional classifiers for each attribute of interest. We provide some brief experiments exploring some major differences between these models, such as using an auxiliary property predictor, the weakly supervised case using said property predictor, and an additional adversarial loss.

We also release code for our shape dataset, which can be generated from a script in our repository with tunable parameters for the occurrence rate of each feature. This dataset, described below, produces images of simple shapes with many known, labeled variables. The key property of this dataset is that one natural language feature, the color, contains over 950 labels, and is the main subject of our experimentations. It is most similar to the dSprites dataset [19], which also produces synthetic shapes for disentanglement. However, our dataset contains natural language descriptions of colors and numerous additional complex features and shapes. The lack of definitive feature labels and the large resolution prevented us from exploring other traditional disentanglement datasets like FFHQ [14]. Because we primarily used this dataset for color-replacement experiments, we designed specific metrics for color-replacement evaluation which were attuned. Other methods exist to evaluate disentangled representations, such as DCI [7], SAP [16], and MIG [5], which cover roughly the same metrics of disentanglement and informativeness as our methods.

3 Dataset

The dataset we use is a custom algorithmically generated set of images of shapes. These images vary in primary shape (3- to 8-sided polygons, stars, and arrows), color, scale, rotation, skew, translation, hatching, and the inclusion/exclusion of a shadow. These features all occur independently with a different probability per feature. The exact generation procedure can be found in the attached GitHub repository [3]. Our goal is to provide a simple dataset for preliminary disentanglement results, which can later be modified for other important factors such as feature correlation and noise.

The intent was to create a large set of properties such that we could supervise a subset of these properties and allow the model to discover the other properties, simulating the desired real-world setting in which some (but not all) of the latent space is specified by the model builder. Examples of generated images can be found in Figure 3.

To extend this to the language supervision task, our generation process samples from the xkcd colorset, which contains the names of 954 different colors based on natural language usage. This large, discrete set of colors presents a unique challenge that finds a natural solution in continuous language embeddings. Additionally, recoloring allows for simple metrics on structural retention and recoloring accuracy.

4 Methods

4.1 Vanilla $\beta-$VAE

The default VAE used as a baseline in this work is a CNN which takes in an input of size $64 \times 64 \times 3$, performs four applications of strided convolution/batch normalization / ReLU until predicting

[3] https://github.com/thomaslu2000/vae-latent-text-alignment

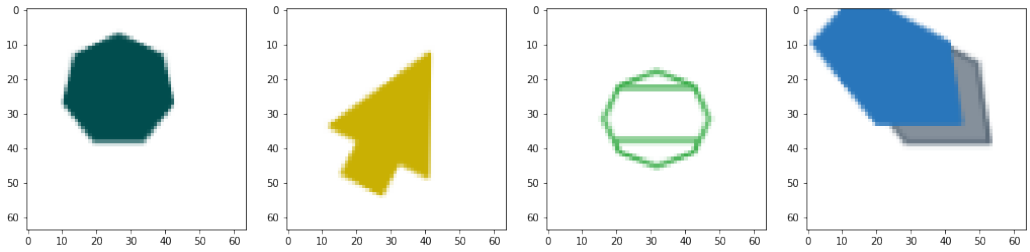


Figure 3: Random generated shapes from the dataset, showing color, shape, hatchedness, shadow, and skewness

the means and standard deviations of a latent vector. Then, similarly, four applications of strided deconvolution/batch normalization / ReLU are applied (sigmoid at the last layer) to reproduce the input image. The loss function is, as in Higgins et al.[11]

$$\beta D_{KL}(N(\mu, \sigma)||N(0, 1)) + E[(X_{recon} - X)^2]$$

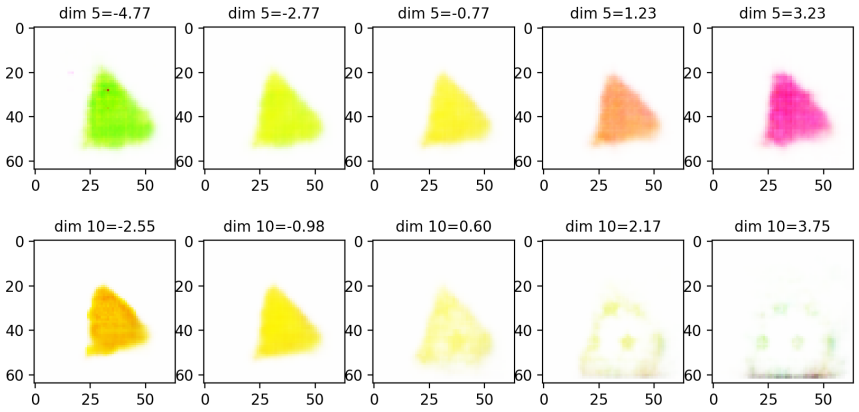


Figure 4: Chosen dimensions of $\beta = 10$ VAE roughly corresponding to Red-Green axis (top) and Hatchedness (bottom). Note that dimensions were found by hand and correlate with other features.

The $\beta-$VAE is capable of some levels of disentanglement over each dimension, as shown in figure 4. However, which dimension controls which feature is unknown, and each dimension may affect multiple features. The task at hand is to train a model such that our target feature is in a known section of the latent space and can be manipulated to any other value without affecting the other properties of the image.

4.2 Language-based Supervision

For our experiments, we embed the natural language phrase for each color using encoders such as USE [4], CLIP [23], or Word2Vec [20]. In general, CLIP embeddings led to more accurate recoloring compared to the other language embedding methods. For USE and CLIP, we were able to experiment with multiple prompts to obtain the language embedding. The prompt leading to the most accurate recoloring was found to be "This is a BLANK colored shape."

4.3 SLt-VAE: Supervised Latent VAE Representations

Our first baseline strategy is an adaptation of the strategy proposed by Van et al.[28] to ensure that our hidden vector z can be linearly mapped to the ground-truth properties about the input image, y: in this case, the language embedding. First, as with a standard VAE, a latent encoding z is obtained from the input image. We then train a linear matrix W such that $\hat{y} = Wz$ estimates the supervised

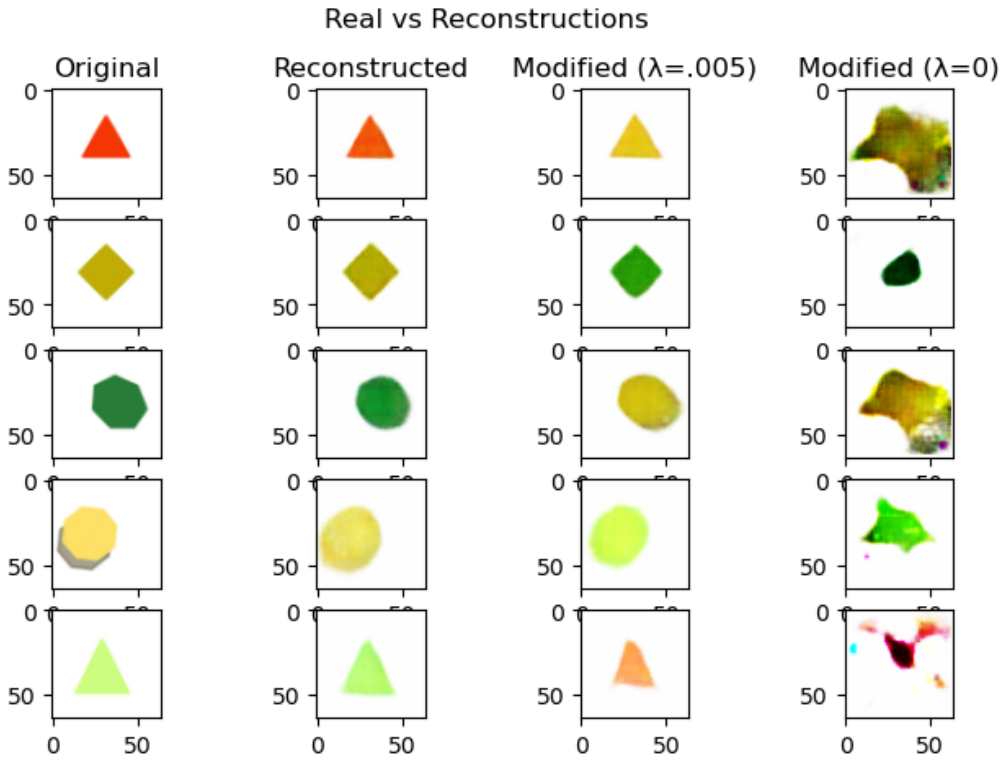


Figure 5: SLt-VAE reconstructions trained on CLIP embeddings. The smoothing parameter drastically contributes to reasonable counterfactual images.

features y. The entire loss we minimize is:

$$E_\epsilon[\alpha(Wz - y)^2] + \beta D_{KL}(N(\mu, \sigma^2), N(0, 1)) + E_\epsilon[(X_{recon} - X)^2]$$

Where z and ϵ are obtained from σ and μ as per a standard VAE. α and β are tune-able hyperparameters. A diagram of this method is shown in 2.

At evaluation time, we wish to minimally modify the latent representation while maximally conforming to a desired color, so we minimize:

$$\|W(\mathbf{z} + \mathbf{a}) - \mathbf{y}\|_2^2 + \lambda \mathbf{a}^T \mathbf{a}$$

In this case, $\mathbf{y}$ is the desired text embedding, $\mathbf{z}$ is the original image's latent representation (usually not resampled at test time), W is the learned matrix which maps from $\mathbf{z}$ to $\mathbf{y}$, and λ is a tune-able hyperparameter for smoothness. We feed the vector $\mathbf{z} + \mathbf{a}$ to the decoder to produce the recolored image. The components orthogonal or along $\mathbf{a}$ will be disentangled representations of the general image and the change in color respectively. It can be shown that the closed-form solution to find $\mathbf{a}$ is as follows. A proof is provided in the appendix.

$$\mathbf{a}^T = (W^T W + \lambda I_d)^{-1} W^T (\mathbf{y} - W\mathbf{z})$$

We found that the smoothing term λ was necessary and we could not simply use the pseudoinverse of W, as this had extremely large eigenvalues which pushed $\mathbf{z} + \mathbf{a}$ far outside of its typical distribution (from typical norms of around 4 to around 11) and led to inaccurate results. The results of recolorings with and without this smoothing term are shown in figure 5.

We also note that SLt-VAE can be extended very naturally to a case of partial supervision. If textual supervision is not available for some samples, that loss term can simply be omitted and the rest of the

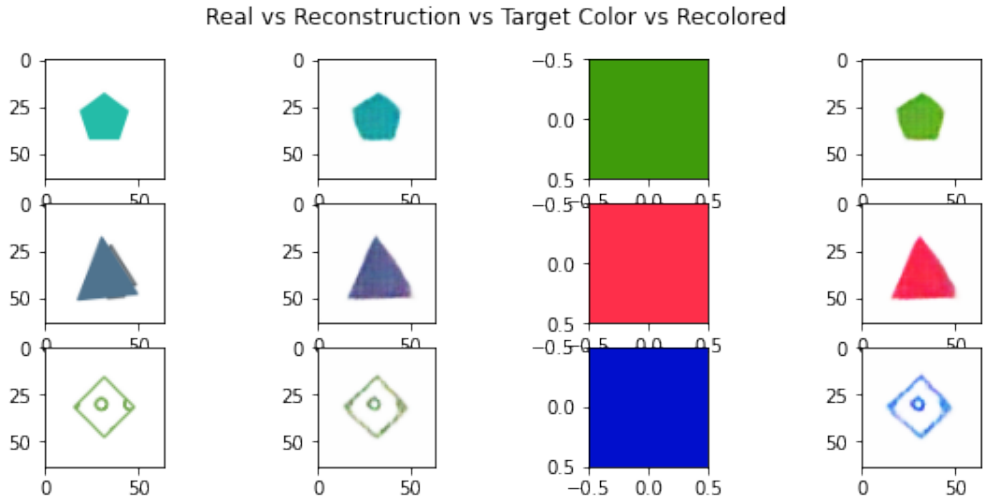


Figure 6: Feature VAE reconstructions for the model trained on CLIP embeddings. The model preserves details such as shape and fill styles while changing the shape to the target color.

model can run as normal – this is not quite as trivial for Feature VAE. Future research might explore reducing the number of samples for which supervision is available.

4.4 Feature VAE: Appending Controllable Parameters to Latent Representation

Our second strategy is to simply concatenate the supervised features y (i.e. the language embeddings) to the latent vector z as an input to the decoder. This method does not directly force the latent space to align with the supervised features. As the dimensionality of the latent space is kept low and the KL divergence loss encourages minimum information to be passed through the latent vector bottleneck, the network is incentivized to not redundantly encode the information present in y.

As a result, we hope that we can control the properties encoded in y by inputting a different y obtained from the language encoder and keeping z constant. The loss is the same as in the vanilla VAE. No additional terms exist to prevent the content of y from being encoded in z in the base model, though we experiment with an adversarial loss against predicting y from z in later experiments. A diagram of this model and certain additional variants are shown in Figure 2. Examples of recoloring using this method are shown in Figure 6.

4.5 Additional Experiments and Comparisons

Adversarial Disentangling One goal is to ensure the latent representation of the shape is independent of color. Similar to many previous works, we train an adversary that attempts to predict color from this extracted representation, with the goal color being unpredictable from the latent space. Therefore, our generator must learn to use the concatenated input for color. We use a gradient reversal layer for adversarial training. At each iteration of the main model updates, we allow the adversary additional optimization steps to learn color from the latent vectors as another hyperparameter. Our experiments covered the adversary attempting to predict either the RGB values or the text embedding itself.

Weak Supervision and Submodel Prediction Because we suspected our model had difficulty learning color from text embeddings compared to RGB values, we were interested in simplifying this concatenated input. Thus, we wanted to explore the weakly supervised case, where we have access to a small proportion of RGB inputs in the training set in addition to language. The test set would only contain language. We wished to see if directing our concatenated inputs with this information would encourage the model to separate the structural information from the color information in each embedding.

Model	Latent Dimension	Color Similarity	Image Similarity		
		Delta-E	PSNR	SAM	SRE
Feature VAE + Word2Vec	50	63.98 ± 0.69	52.78 ± 0.17	89.48 ± 0.28	57.30 ± 0.14
Feature VAE + USE	50	64.73 ± 0.69	52.81 ± 0.16	89.91 ± 0.02	57.21 ± 0.13
Feature VAE + CLIP	20	81.18 ± 0.56	52.91 ± 0.18	89.92 ± 0.02	57.13 ± 0.13
Feature VAE + CLIP	100	79.41 ± 0.62	52.54 ± 0.16	89.93 ± 0.01	56.74 ± 0.12
Feature VAE + CLIP + Adversarial	50	82.32 ± 0.50	51.30 ± 0.18	89.92 ± 0.02	56.37 ± 0.13
Feature VAE + CLIP + 25% W.S.	50	88.30 ± 0.35	52.91 ± 0.17	89.92 ± 0.02	57.34 ± 0.13
Feature VAE + CLIP + 50% W.S.	50	88.84 ± 0.33	53.65 ± 0.18	89.91 ± 0.02	57.49 ± 0.13
SLt-VAE + USE	16	60.47 ± 0.69	52.67 ± 0.19	89.80 ± 0.16	57.06 ± 0.13
SLt-VAE + USE	64	60.35 ± 0.70	53.06 ± 0.17	89.80 ± 0.16	57.31 ± 0.13
SLt-VAE + CLIP	16	59.08 ± 0.72	52.85 ± 0.19	89.80 ± 0.16	57.06 ± 0.13
SLt-VAE + CLIP	64	60.66 ± 0.72	53.01 ± 0.17	89.84 ± 0.13	57.15 ± 0.13

Table 1: Validation metrics of supervised VAE variants at a fixed value of $\beta = 1.0$. Feature VAE yields the closest-matching recolorings, while SLt-VAE and Feature VAE score similarly according to structural similarity metrics. Adversarial disentangling and weak RGB supervision further improve the performance of FeatureVAE.

With this method, we train a sub-model to predict RGB from text embeddings, and then, during testing, we append this prediction to the latent vector rather than the text embedding. In our training process, we first pretrain the RGB prediction model on known colors with MSE loss. Then, we jointly train our Feature VAE and RGB prediction models on reconstruction, with an additional MSE loss for predicting known RGB values. If RGB values are known, those are fed to the generator instead of predictions. We experimented with beginning training with examples with known RGBs and then increasing the number of unknown examples per epoch at various rates.

5 Results

A summary of our results can be found in Table 1 using the quantitative metrics designed for our dataset. These metrics are described below. We performed numerous experiments while sweeping over a number of hyperparameters. The full results are of our hyperparameter sweep are available in our supplementary submission.

For the SLt-VAE, we swept the dimensionality of the latent vector over a range of 8 to 64, the β parameter over a range of 0.5 to 2.5, smoothness over a range of 0.001 to 0.005, and α over a range of 0.1 to 2.5.

For the Feature VAE, we swept over the dimensionality of the latent vector over a range of 5 to 100, and the β parameter over a range of 0.01 to 5. When an additional RGB predictor was enabled, additional parameters included the mixing proportion (the proportion of labeled data), the temperature parameter for using labeled data (controls the rate at which the model uses true labels when available), and the lambda weight of the RGB predictor loss. When adversarial methods were enabled, there was an additional sweep over the number of adversary steps per generator step, and the lambda weight of the adversary loss.

5.1 Quantitative Metrics

We leverage powerful structural and color-theoretic metrics of reconstruction quality estimation for a powerful assessment on this novel task in image perturbation [22, 21].

1. **Reverse Scaled Delta-E Score** The Delta-E (dE) is a metric that serves as a representation of the Euclidean "distance" between two colors. Scale [100: Good (no color difference), 0: Bad (high color difference)]

2. **ISSM** The Information theoretic-based Statistic Similarity Measure (ISSM) combines the principles of information theory with statistics for establishing relationships among image intensity values. Scale [1: Good, 0: Bad][1]

3. **PSNR** The Peak Signal-to-Noise Ratio (PSNR) quantifies the relationship between the maximum achievable power of the ground truth and the power of corrupting noise that impacts the accuracy of its reconstruction. PSNR is commonly denoted on the logarithmic decibel scale. Scale [Above 40 dB: Good, Below 40 dB: Bad][12]

4. **RMSE** The Root Mean Square Error (RMSE) quantifies the magnitude of per-pixel variation resulting from the reconstruction task. RMSE values are always non-negative, with a value of 0 indicating complete similarity between the compared ground truth and reconstruction. Scale [0: Good, ∞: Bad][25]
5. **SAM** The Spectral Angle Mapper assesses the spectral resemblance of two spectra by computing the angle between them, treating the spectra as vectors in a multidimensional space where the dimensionality corresponds to the number of bands. Scale [0 or small angles: Good, Large angles: Bad][30]
6. **SRE** The Signal to Reconstruction Error ratio quantifies the reconstruction error relative to the power of the ground truth. SRE is commonly denoted on the logarithmic decibel scale. Scale [Above 5 dB: Good, Below 5 dB: Bad][17]

RMSE and ISSM are 0 for all experiments. Delta-1 is the score measured between image and color, and Delta-2 is the score measured between images.

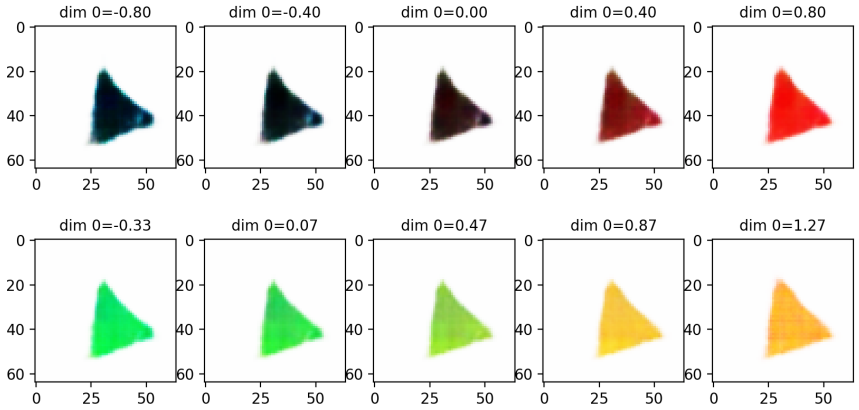


Figure 7: Feature VAE recoloring of a given shape under two chosen colors, while varying the "Red-ness" dimension (assigned to dimension 0).

6 Discussion and Analysis

Based on our experiments, we provide the following key findings from this study:

1. For one additional experiment, we used direct RGB values as opposed to language embeddings. As expected, our models often perform worse on color matching using language embeddings as opposed to our RGB baselines. This is likely because it must predict the color from pretrained language embeddings rather than being able to directly use simple ground truth color values.
2. Models weakly supervised with correct RGB models, which were able to produce regressors for RGB values, performed better than models without it. This suggests that models such as LORD benefit from their auxiliary classifiers provided they produce correct classifications.
3. The CLIP Embeddings have the best performance when compared to Word2Vec and USE. This is consistent with existing literature on text supervision for image generation, where CLIP embeddings are a canonical choice due to their image supervision and in other disentanglement tasks.[24] [8]
4. Color matching performance appears to have a strong negative correlation with the dimensionality of the model and a strong positive correlation with the β parameter of the model. We hypothesize that this is because weaker information theoretic constraints on the latent space allow our models to “hide” color information within the latent vector in irregular ways that cannot be captured with our methods.
5. Structural similarity appears to have a strong positive correlation with the dimensionality of the model and a strong negative correlation with the β parameter of the model. This is likely because weaker information theoretic constraints allow more structural information to be encoded in the latent vectors.

6. SLt-VAE is not very sensitive to any hyperparameters except smoothing, which is very cheap to tune.
7. Feature VAE models perform worse on color matching when information theoretic constraints are weakened, such as increasing latent vector size and lowering the value of β. This is likely due to the model storing color information in the latent vector rather than learning it from the language embedding. Thus, this method works best with sufficient constraints.
8. Our adversarial methods did not perform significantly stronger than our standard models. However, this could be due to the simplicity of the problem, or from the difficulty of training stable adversarial models.
9. Feature VAE was capable of recoloring shapes to colors outside of the dataset, while SLt-VAE was unable to do so. Feature VAE extracts color through the decoder, and thus is likely more generalizable than SLt-VAE, which extracts color only through the linear transform W.

7 Conclusion

In this work, we propose a novel benchmark dataset and results for disentangling representations of variational auto-encoders for text-conditioned image generation. We propose and evaluate strategies for supervision and control through parametric interventions and embedding conditioning. The key takeaways of this study suggest possible barriers (color signal extraction) and useful methods (weak supervision) towards effective disentangling for future work. Motivated by our findings, we consider improving this model by improving our ability to extract a relevant structural signal from the prompt, using methods such as a richer training set of text inputs, and extending beyond this dataset towards natural imagery in future work. We hope that this study contributes towards a better understanding of disentangling representations in the latent space for better control and supervision in future generative models.

References

[1] Mohammed Abdulameer Aljanabi, Zahir M Hussain, Noor Abd Alrazak Shnain, and Song Feng Lu. Design of a hybrid measure for image similarity: a statistical, algebraic, and information-theoretic approach. *European Journal of Remote Sensing*, 52(sup4):2–15, 2019.

[2] Jinwon An and Sungzoon Cho. Variational autoencoder based anomaly detection using reconstruction probability. *Special lecture on IE*, 2(1):1–18, 2015.

[3] Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in beta -vae. *arXiv preprint arXiv:1804.03599*, 2018.

[4] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*, 2018.

[5] Ricky TQ Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. *Advances in neural information processing systems*, 31, 2018.

[6] Nat Dilokthanakul, Pedro AM Mediano, Marta Garnelo, Matthew CH Lee, Hugh Salimbeni, Kai Arulkumaran, and Murray Shanahan. Deep unsupervised clustering with gaussian mixture variational autoencoders. *arXiv preprint arXiv:1611.02648*, 2016.

[7] Cian Eastwood and Christopher K. I. Williams. A framework for the quantitative evaluation of disentangled representations. In *International Conference on Learning Representations*, 2018.

[8] Aviv Gabbay, Niv Cohen, and Yedid Hoshen. An image is worth more than a thousand words: Towards disentanglement in the wild. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 9216–9228. Curran Associates, Inc., 2021.

[9] Aviv Gabbay and Yedid Hoshen. Demystifying inter-class disentanglement. *arXiv preprint arXiv:1906.11796*, 2019.

[10] Naama Hadad, Lior Wolf, and Moni Shahar. A two-step disentanglement method. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 772–780, 2018.

[11] 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. In *International conference on learning representations*, 2017.

[12] Alain Hore and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th international conference on pattern recognition*, pages 2366–2369. IEEE, 2010.

[13] Tom Joy, Sebastian M Schmon, Philip HS Torr, N Siddharth, and Tom Rainforth. Capturing label characteristics in vaes. *arXiv preprint arXiv:2006.10102*, 2020.

[14] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

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

[16] Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational inference of disentangled latent concepts from unlabeled observations. *arXiv preprint arXiv:1711.00848*, 2017.

[17] Charis Lanaras, José Bioucas-Dias, Silvano Galliani, Emmanuel Baltsavias, and Konrad Schindler. Super-resolution of sentinel-2 images: Learning a globally applicable deep neural network. *ISPRS Journal of Photogrammetry and Remote Sensing*, 146:305–319, 2018.

[18] Michael F Mathieu, Junbo Jake Zhao, Junbo Zhao, Aditya Ramesh, Pablo Sprechmann, and Yann LeCun. Disentangling factors of variation in deep representation using adversarial training. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

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

[20] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[21] Markus U Müller, Nikoo Ekhtiari, Rodrigo M Almeida, and Christoph Rieke. Super-resolution of multispectral satellite images using convolutional neural networks. *arXiv preprint arXiv:2002.00580*, 2020.

[22] Ph.D. Niku Ekhtiari. Comparing ground truth with predictions using image similarity measures, 2021.

[23] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[24] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *CoRR*, abs/2112.10752, 2021.

[25] Umme Sara, Morium Akter, and Mohammad Shorif Uddin. Image quality assessment through fsim, ssim, mse and psnr—a comparative study. *Journal of Computer and Communications*, 7(3):8–18, 2019.

[26] Stanislau Semeniuta, Aliaksei Severyn, and Erhardt Barth. A hybrid convolutional variational autoencoder for text generation. *arXiv preprint arXiv:1702.02390*, 2017.

[27] Jakub Tomczak and Max Welling. Vae with a vampprior. In *International Conference on Artificial Intelligence and Statistics*, pages 1214–1223. PMLR, 2018.

[28] Toan Pham Van, Tam Minh Nguyen, Ngoc N. Tran, Hoai Viet Nguyen, Linh Bao Doan, Huy Quang Dao, and Thanh Ta Minh. Interpreting the latent space of generative adversarial networks using supervised learning. In *2020 International Conference on Advanced Computing and Applications (ACOMP)*, pages 49–54, 2020.

[29] Chuhan Wu, Fangzhao Wu, Sixing Wu, Zhigang Yuan, Junxin Liu, and Yongfeng Huang. Semi-supervised dimensional sentiment analysis with variational autoencoder. *Knowledge-Based Systems*, 165:30–39, 2019.

[30] Roberta H Yuhas, Alexander FH Goetz, and Joe W Boardman. Discrimination among semi-arid landscape endmembers using the spectral angle mapper (sam) algorithm. In *JPL, Summaries of the Third Annual JPL Airborne Geoscience Workshop. Volume 1: AVIRIS Workshop*, 1992.

[31] Muhan Zhang, Shali Jiang, Zhicheng Cui, Roman Garnett, and Yixin Chen. D-vae: A variational autoencoder for directed acyclic graphs. *Advances in Neural Information Processing Systems*, 32, 2019.