Noise as Masks

Lingyu Zhang

Abstract

Motivated by the success of masked language models [2], many recent efforts have gone into masked image modeling for self-supervised learning in vision [1, 3]. However, it is not obvious how to design masks that can lead to optimal performance. While randomized patch masking have shown great potential, more recent approaches suggest semantic-guided [5] and fine-grained masking [7] can be used for learning more useful representations. In this work, we analyze the limitation of these masking methods and propose an alternative: use additive noise to replace masking. The idea is based on the fact that injecting noise induces uncertainty, which can be used for fine-grained information control. Preliminary empirical results support our approach.

1. Introduction

Recent advances in self-supervised learning have explored the possibility to pretrain scalable vision models with masked reconstructive tasks [1, 3]. However, it is not obvious how to design masks that can lead to optimal performance. Many approaches use randomized patch masking, while more recent works [5, 7] argues that *what is masked* is more crucial than *how much is masked*. To enable masking of whole entities, [7] proposed an adversarial masking method, where a pixel-level learnable mask aims at occluding what's semantically important, showing improvement in learned representations. However, learning masks requires propagating gradients through a non-differentiable hard-assignment operation. Using gradient approximation methods such as straight-through estimators results in suboptimal performance. The authors of [7] use real-valued masks instead: learned masks with continuous values between $[0, 1]$ are multiplied with the image using the Hadamard product. We argue that real-valued continuous masks are fundamentally different from binary masks because smaller mask values doesn't necessarily mean lower amount of information flow, *i.e.* information monotonicity is not guaranteed. Intuitively, it is unstable to control information by reducing pixel intensity. We show this in section **??**. We thus propose an alternative way to perform fine-grained pixel level information control over an image — by injecting noise. The intuition is that adding noise to a pixel value induces uncertainty over that value, which is theoretically equivalent to removing information, resulting in a parameterized soft-occlusion model.

Table 1. Comparison of different masksing

Method	Properties		
	Semantic	Pixel-level	Monotonic
BEiT [1]	✗	✗	✓
MAE [3]	✗	✗	✓
SEMMAE [5]	✓	✗	✓
AIDOS [7]	✓	✓	✗
Proposed	✓	✓	✓

2. Method

Given an image, we would like to induce controllable uncertainty over the pixels and allow learnable "distribution of uncertainty" over pixels. One way to

A natural way to quantify uncertainty is the Shannon entropy. For

2.1. Implementation

We propose to learn a "mask" $m = \mathcal{M}(x)$ from an image x whose entries are the variance σ^2 of a zero-mean Gaussian at every pixel location, which we add to the original image to obtain a noise-occluded view. we use the reparameterization trick to enable backpropagation. We adopt a similar approach to [7], using a U-Net architecture for parameterizing $\mathcal{M}$. The noise-occluded image can then be passed through an encoder $\mathcal{E}$ for some predictive task. We optimize for some desired objective by updating the mask model's weights. By doing so, we learn a noise based occlusion model that learns what to mask in the pixel space. The higher the variance of noise on a pixel is, the less information it contains and therefore is less necessary.

This can be applied to a variety of tasks, since it provides the ability to perform pixel-wise differentiable information control. Potential tasks to work on: (1) Adversarial masking based contrastive learning as in [7]; (2) Entity-wise multi-modal contrastive learning (pairing masked words with masked visual entities); (3) Interpretable AI and identifying spurious correlations; (4) Sparse adversarial attacks (spending attack budget on semantic areas); (5) Video block masking.

This can be measured in terms of entropy. A deterministic pixel value has $E_{det} = -\sum_x p(x)\log p(x) = -1 \cdot 0 = 0$, where a uniform distribution over 256 pixel values have $E_{uni} = -\sum_x p(x)\log p(x) = 256 \cdot \frac{1}{256} \log \frac{1}{256} = 8$ bits.

We propose to learn a "mask" $\mathcal{M}(x)$ from an image x that outputs the variance σ^2 of a zero-mean Gaussian at every pixel location, which we add to the original image to obtain a noise-occluded view. we use the reparameterization trick to enable backpropagation. We adopt a similar approach to [7], using a U-Net architecture for parameterizing $\mathcal{M}$. The noise-occluded image along and the original image are encoded into z_m and z with an encoder $\mathcal{E}$, which can be any vision backbone. We then solve for the following objective:

$$\mathcal{M}^*, \mathcal{E}^* = \arg\min_{\mathcal{E}} \arg\max_{\mathcal{M}} \mathcal{L}_{ssl}(z, z_m) \tag{1}$$

Where $\mathcal{L}_{ssl}$ is an arbitrary contrastive loss. Our setup is very similar to that of [7], except that our masking function is now a noise-based soft occlusion model.

3. Analysis

We propose to use noise in place of binary or real-valued masks for learning occlusion models. Binary masks are not strictly differentiable and real-valued masks have suboptimal control over information. This is because real-valued masks learn to reduce pixel intensity, but there are neither theoretical nor empirical guarantees that this can actually reduce information. Darkening regions could possibly lead to higher recognizability (such as for shadows). Noise with learned variance, on the other hand, circumvents both these caveats. Here we provide some formal analysis and intuition.

Consider the original image as a fixed point in the pixel space. For every spatial location, we add a zero-mean gaussian with learned variance. We chose gaussians for their well-known analytical properties. After the "masking", it becomes a high-dimensional gaussian centered at the original location. The shape of the gaussian is learned, and is determined by the "importance" of each dimension, *e.g.* pixels of an image. The more important a dimension is, the lower variance it should have in its values. As shown in Fig. 1, real-valued masks only move data points closer to the origin, while noise masks create a region of uncertainty. The only reason why real-valued masks can reduce information is because data can only be stored to finite precision in computers, decreasing the pixel value limits the total number of values it can take on.

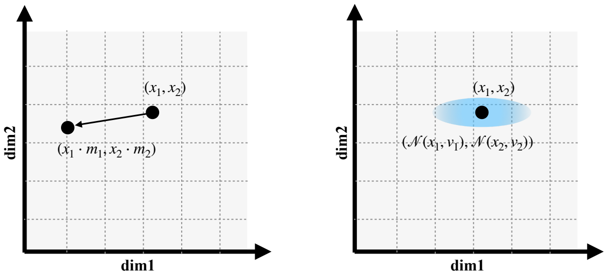
(x_1, x_2)
$(x_1 \cdot m_1, x_2 \cdot m_2)$
dim2
dim1
(x_1, x_2)
$(\mathcal{N}(x_1, v_1), \mathcal{N}(x_2, v_2))$
dim2
dim1

Figure 1. Real-valued masks (left) only moves data in the pixel space, without reducing information. Adding noise (right) induces uncertainty.

In order for noise masked images to be maximally informative, the trivial case would be to just learn to have zero variance everywhere, reverting back to the degenerate case of deterministic data, therefore regularization is necessary. A natural regularization would be the entropy of the gaussians. However, we chose the continuous gaussian because the reparamterization trick allows us to easily perform back propagation, and the entropy of continuous distributions tends to be infinite when discretized. The differential entropy of continuous gaussian is easy to compute but has undesirable properties such as it can be negative. For now, we use a clipped version:

$$\begin{aligned} H(x) &= \frac{D}{2}(1 + \log(2\pi)) + \frac{1}{2}\log|\Sigma| \\ &= \frac{D}{2}(1 + \log(2\pi)) + \frac{1}{2}\sum_{i=0}^{D-1} max(\log(\text{Var}(x_i)), -10) \end{aligned} \tag{2}$$

We also experimented with using a categorical distribution over 256 pixel values and differentiate with the Gumbel-max trick [4], but it resulted in unstable training.

We draw some connections with variational inference. Learning noise occluded images is assuming that every image is a data point x sampled from an underlying distribution – a distribution describing the essential content in the image that is relevant to the prediction target. For example, when classifying an image of a dog, the pixels within the dog should be similar in each sample, while the background pixels can vary to a large degree and still remain semantically similar. The masking model would be approximating the posterior distribution $p(z|x)$. We then sample from this distribution to perform the actual prediction task. We regularize the the posterior by the noise entropy of the encoder $H(z|x)$.

If the prediction target is the original image itsself, then it becomes similar to a VAE, except that we don't regularize the posterior to be close to a standard gaussian. The goal of our encoder (masking model) is not to directly reduce the data to abstract concepts, but to outline the parts of the data that are related to those concepts so that the downstream encoder can learn better representations. Therefore, the other difference is our latent variable z has the same dimension as x.

Our framework is also related to the information bottleneck.

$$\begin{aligned} IB &= I(z;y) - \beta I(z;x) \\ &= I(z;y) - \beta(H(z) - H(z|x)) \\ &= I(z;y) + \beta H(z|x) - \beta H(z) \end{aligned} \tag{3}$$

While the first two terms are identical to ours, we do not regularize the entropy of the marginal noisy image distribution.

4. Toy Experiment

We construct the following toy experiment to provide some intuitions. Consider a binary classification task for two-dimensional data x. Let x_0, x_1 be the two dimensions. The labels y are uniformly sampled from $\{0, 1\}$. x is then generated by the following process:

$$\begin{aligned} p(x_0|y=0) &= \mathcal{U}(0,5) \\ p(x_0|y=1) &= \mathcal{U}(5,10) \\ p(x_1) &= \mathcal{U}(-10,0) \end{aligned} \tag{4}$$

In words, dimension 0 takes on values in $(0, 10)$ and dimension 1 takes on values between $(-10, 0)$. Dimension 0 is the only predictive dimension, dimension 1 is independent of the labels. When dimension 0 takes on values between $(0, 5)$, the ground truth label is 0, when it takes on values between $(5, 10)$, the ground truth label is 1. We further make the task more difficult by randomly switching the two dimensions before prediction. This is more realistic because for image data, the predictive dimensions vary among images.

We use 2-layer MLPs for the noise model $\mathcal{M}$ and the classifier $\mathcal{C}$. The data is passed through the noise model before inputted into the classifier:

$$x \xrightarrow{\mathcal{M}} \tilde{x} \xrightarrow{\mathcal{C}} \hat{y} \tag{5}$$

We regularize by forcing the learned noise on each data point to have a fixed entropy. We expect our masking model to first learn to recognize the irrelevant dimension , and allocate more variance to that dimension. This is possible because we constructed the data in a way that the dimension with negative values are always the irrelevant one. As shown in Fig. 2 and Fig. 3, we plot the ground truth decision rule, the learned classification results, and the shape of learned gaussians. For every data point, the dimension that has negative values is parallel to the decision boundary, therefore contains no information for prediction.

From the plot we can easily see the pattern in the shape of the learned gaussians. As we have discussed previously, for every data point, the dimension that has negative values is the one that is irrelevant for prediction. We can observe in the figure that the model learns to assign large variance to the dimension that has negative values, allowing it to vary while remaining accurate in prediction.

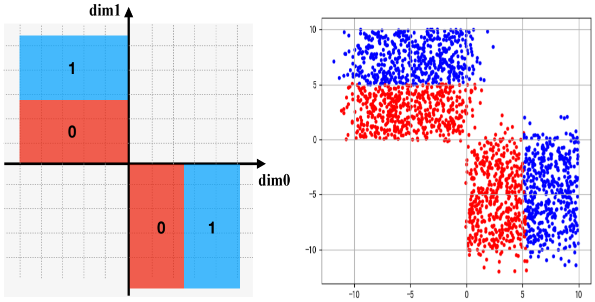

Figure 2. The ground truth decision rule (left); the learned classification results (right).

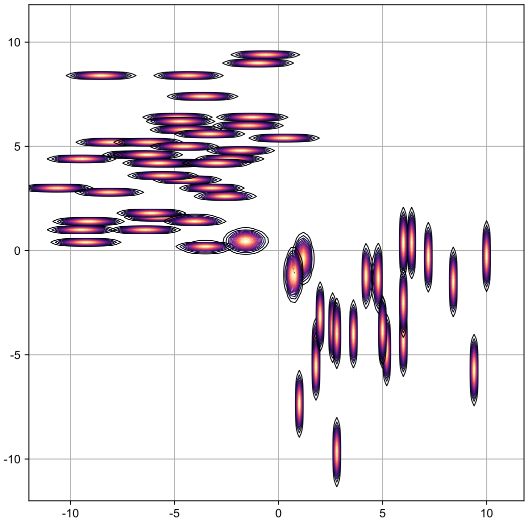

Figure 3. Ellipses of learned gaussians

Adding noise to images before prediction is not uncommon. Adding noise with fixed uniform variance is a common data augmentation and can be shown to be equivalent to ridge regression. Our learned noise, on the other hand, learns to be obfuscate necessary regions and is more flexible. We plot the test accuracy using learned noise and fixed noise with the same entropy Fig. 4. It is clear that learned noise adapts to the data better, learning to preserve predictive dimensions. Note that since in our experiment, the training set and test where generated from identical distributions, and we provided sufficient training data, adding noise is not expected to help generalize.

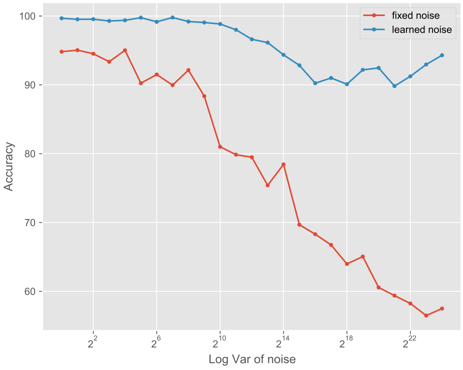

Figure 4

5. Experiments

5.1. Masked Autoencoding

Masked autoencoders [3] have been shown to be a promising direction of self-supervised learning for vision. Ordinary autoencoders cannot learn interesting representations because without further constraints, the encoder-decoder can just learn the identity function in the most trivial way. Masked autoencoders performs autoencoding by first occluding pixels or patches of the image, then regress the original image. By only revealing part of the data and asking the model the regressed the occluded parts, it is required to reason about relationship between the data parts and therefore learn useful representations by discovering the structure within.

There are many ways to occlude fractions of images. The first work to implement this idea was [8], where input pixels where randomly set to zero by a given probability, which governs the corruption level. Similarly, [1,3] propose to exclude image tokens of an autoencoding transformer. More recently, [7] proposed learning real-valued multiplicative per-pixel semantic masks leads to better representations.

In this section, we describe our method applied in masked autoencoding for self-supervised learning. We also compare the performance using various masking schemes. We evaluate the quality of learned representations by the standard linear probing. We follow [8], choosing an 2-layer MLP with neurons 784-2000-2000 for the encoder architecture. The decoder is symmetric to the encoder. As suggested by in the original paper, using a wide bottleneck layer ensures learning over-complete representations. As shown in Tab. 2, using learned noise as masks outperforms other methods.

5.2. Saliency Detection

We experiment with noise masking under two settings. **Negative mask:** learning to mask out predictive entities by maximizing embedding distance of masked and original image; **Positive mask:** learning to mask out irrelevant regions by minimizing embedding distance of masked and original image.

Table 2. Comparison of different masking

Method	Error Rate lin
Supervised	1.76
Raw input	7.58
AE	7.90
Denoising AE	2.38
Learned Noise (Proposed)	**1.75**

Table 3. Raw input is just directly applying logistic regression on input images without any feature extraction.

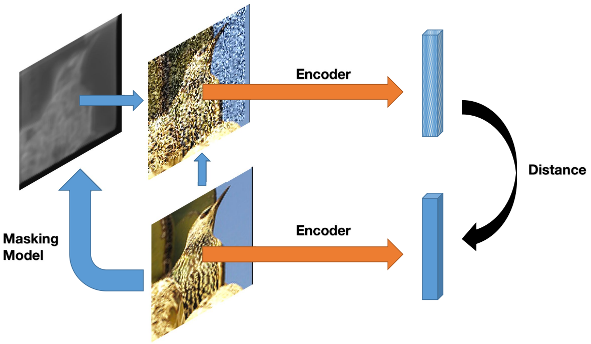

Figure 5. **Overview.** We first use a U-Net like masking model to generate a map of gaussian parameters. We then sample from the gaussians and add to the original image. Both noisy image and original image is then passed through the same pretrained encoder. Determining on whether we are masking predictive or irrelevant regions, we maximize or minimize the embedding distance.

We used a pretrained resnet18 as encoder, and compute NT_Xtent loss between noising image and original image embeddings. We use a state-of-the-art U2NET [6] for learning masks. For negative masking, we maximize the embedding distance while minimizing the noise entropy. This can be seen as adversarial masking where one tries to mask out predictive pixels with as small amount of noise as possible. We do the opposite for the positive mask, where embedding distance is minimized and noise entropy is maximized. Intuitively, this is trying to add as much noise to the image as possible without changing its semantic content. We visualize the learned mask in Fig. 6

From these experiments, we show that it is possible to use embedding geometry as self-supervion for learning entity-wise relations. For the next step, we plan to apply learned noise as augmentation for common self-supervised frameworks such as SimCLR.

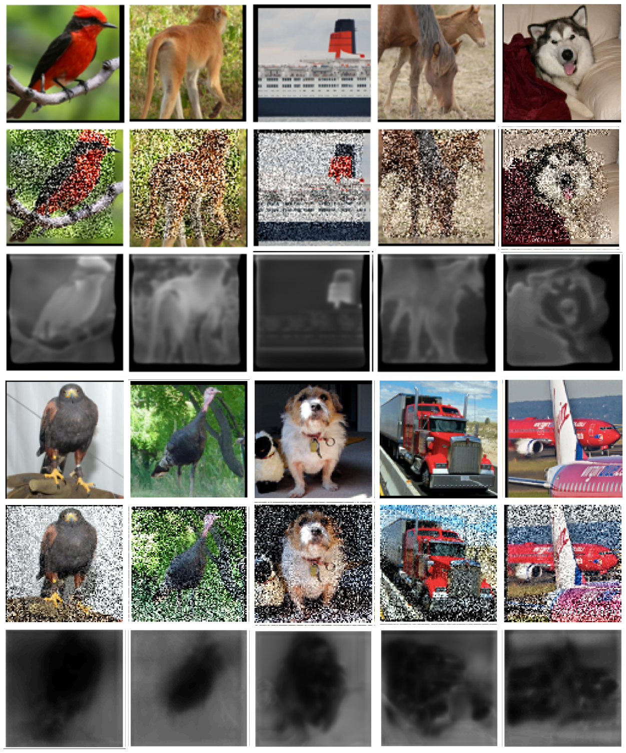

Figure 6. Visualization of learned noise masks. Row 1, 2, 3 are negative masking: predictive entities have been covered with noise with large magnitude; Row 4, 5, 6 are positive masking: irrelevant regions have been covered with noise leaving the category-relevant entity recognizable.

6. Open Questions

6.1. Can the mean of the gaussians be learned too?

There's no reason to believe pixels in the original images are the optimal centers for the noise. Predictive pixels of course should be closer to the best means, but irrelevant pixels are unlikely to be. Allowing the mean to be learned seems to give more flexibility to the masking model, but will this also allow some shortcuts leading to trivial results? Given more parameters, will it be harder to train?

6.2. How to better formulate the regularization

While we addressed one trivial case of all-zero variance by regularizing the entropy, there are other shortcuts. The model can learn to allocate all variance to one pixel, and remain highly predictive since everything else is the same. Currently, we are clamping the maximal variance for single pixels, but it doesn't seem elegant enough. It has also been difficult tuning the hyper-paramters.

References

[1] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021. 1, 4

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 1

[3] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 1, 4

[4] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016. 2

[5] Gang Li, Heliang Zheng, Daqing Liu, Chaoyue Wang, Bing Su, and Changwen Zheng. Semmae: Semantic-guided masking for learning masked autoencoders. *arXiv preprint arXiv:2206.10207*, 2022. 1

[6] Xuebin Qin, Zichen Zhang, Chenyang Huang, Masood Dehghan, Osmar R Zaiane, and Martin Jagersand. U2-net: Going deeper with nested u-structure for salient object detection. *Pattern recognition*, 106:107404, 2020. 4

[7] Yuge Shi, N Siddharth, Philip HS Torr, and Adam R Kosiorek. Adversarial masking for self-supervised learning. In *International Conference on Machine Learning*, 2022. 1, 2, 4

[8] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol, and Léon Bottou. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(12), 2010. 4