Visual Language Modeling on CNN Image Representations

Hiroharu Kato
The University of Tokyo
kato@mi.t.u-tokyo.ac.jp

Tatsuya Harada
The University of Tokyo
harada@mi.t.u-tokyo.ac.jp

Abstract

Measuring the naturalness of images is important to generate realistic images or to detect unnatural regions in images. Additionally, a method to measure naturalness can be complementary to Convolutional Neural Network (CNN) based features, which are known to be insensitive to the naturalness of images. However, most probabilistic image models have insufficient capability of modeling the complex and abstract naturalness that we feel because they are built directly on raw image pixels. In this work, we assume that naturalness can be measured by the predictability on high-level features during eye movement. Based on this assumption, we propose a novel method to evaluate the naturalness by building a variant of Recurrent Neural Network Language Models on pre-trained CNN representations. Our method is applied to two tasks, demonstrating that 1) using our method as a regularizer enables us to generate more understandable images from image features than existing approaches, and 2) unnaturalness maps produced by our method achieve state-of-the-art eye fixation prediction performance on two well-studied datasets.

1. Introduction

Measuring naturalness of images is an important problem. By measuring naturalness, one can generate realistic images or detect unnatural regions in images.

Convolutional Neural Networks (CNNs) [34] extract and abstract features from raw image pixels hierarchically. Because representations they have learned for image classification are highly discriminative and generalized [10, 48], they are an extremely important component in computer vision. However, they are known to be insensitive to the naturalness of images. For example, images generated through them appear to be strange and unrealistic [40]. Furthermore, they are susceptible to unnatural noise or artificial fabrication [16, 41, 51]. Therefore, a method to measure naturalness can be complementary to CNN features.

Despite the importance of measuring the naturalness of images, few alternative methods exist. Although many probabilistic image models have been proposed [1, 19, 29, 33, 45], most are applicable only for small image patches, not for large natural images. Moreover, they are generally built directly on raw image pixels. Although it is a preferred property for low-level image processing such as image denoising, they have insufficient capability of modeling complex and abstract naturalness as we feel.

In this work, we assume that naturalness can be measured by predictability on high-level features during eye movement. For example, during moving of our eyes from left to right, one feels strangeness when viewing a scene that is not predicted by what we have seen along the way. This strangeness is presumably based on edges, shapes, or more semantic signals: not on pixel-level information.

This type of naturalness is studied extensively and is used widely in natural language processing [43]. Such a method, called Language Model (LM), is applied mainly for regularizing outputs of speech recognition or machine translation systems. Given a sequence of words, LM pre-
dicts the next word from past words for each timestep and computes the naturalness of the entire sequence as the product of prediction accuracy of all timesteps. Although traditional n-gram models, which predict the next word from the prior n words, are still prevalent, LM based on Recurrent Neural Networks (RNNs) [39] has emerged as a favorable option because it can predict the next word using more than n words.

Motivated by the description above, we propose a method to evaluate image naturalness by building a variant of RNN Language Models on mid-level image representations extracted from a pre-trained CNN. RNNLM cannot accommodate two-dimensional representations. Therefore, we apply it vertically and horizontally. We designate this method as Visual Language Model on CNN (CNN-VLM). Figure 1 presents an illustration of our method. Figure 2 presents an example of unnaturalness (or prediction error) maps obtained using our method. CNN-VLM measures the naturalness using contextual information related to high-level discriminative features of CNN, which is difficult to accomplish using probabilistic image modeling on raw pixels.

The concept to treat image representations like words is related to the well-known Bag-of-Visual-Words [7, 49]. However, in our case, the representations are not quantized.

We apply our method to 1) image reconstruction from image features and 2) eye-fixation prediction. The naturalness of images plays an important role in these two tasks. We briefly describe them below.

Image reconstruction from image features is a task to visualize an image feature by reconstructing an image from it. To reconstruct interpretable images, regularizers of some kind must be imposed on images [38, 47]. We demonstrate empirically that using CNN-VLM as a regularizer enables us to generate more understandable images than those produced using two existing methods.

Human eye fixation points on images were shown to be predictable using Shannon’s “self-information” [5]. In fact, many attention models are explainable from the perspective of information theory [3]. Because self-information can be interpreted as unlikelihood or unnaturalness, our unnaturalness map is useful as a saliency map. We demonstrate that CNN-VLM achieves state-of-the-art performances on eye fixation prediction task in the experiment section.

Our contributions are summarized as follows. 1) Based on the assumption that naturalness can be measured by the predictability on high-level features during eye movement, we proposed a novel method to evaluate naturalness by building a variant of RNN Language Models on pre-trained CNN representations. 2) We confirmed that using our method as a regularizer enables us to generate more understandable images from features than existing approaches. 3) We showed that unnaturalness maps produced using our method achieve state-of-the-art eye fixation prediction performance on two well-studied datasets.

2. Related work

This section presents a brief review of existing approaches which are related to modeling naturalness of images. Additionally, we describe image reconstruction from features and eye fixation prediction.

2.1. Image modeling

Many probabilistic image models have been proposed [1, 15, 19, 29, 33, 45, 15]. However, most are only applicable to small and fixed-size image patches of simple contents. To overcome this matter, Gregor et al. [18] used attention mechanisms and succeeded in generating very realistic images. Denton et al. [9] applied Generative Adversarial Networks [15] in a hierarchical way.

Theis and Bethge [53] proposed a scalable image model using multi-dimensional LSTMs [17] which predict pixel values of certain locations from preceding pixels. We also use RNN for predictions like theirs. However, to capture more high-level information, we train RNN on CNN representations, not on raw pixels.

Several vision papers explicitly use LM. Wu et al. [58] and Tirilly et al. [54] trained LMs on quantized local descriptors or Visual Words. Although their approach is similar to ours, they used LMs for classification, not for measuring naturalness. Ranzato et al. [42] trained a language model on a small region of videos which predicts the next time frame to learn spatial-temporal video representations.
2.2. Image reconstruction from features

Reconstructing an image from its representation is an important task to understand the characteristics of the representation. Many works have addressed this problem for hand-crafted representations [8, 28, 38, 56, 57] and deep representations [11, 38, 47, 59].

Mahendran and Vedaldi [38] showed that an image can be reconstructed by gradient descent if the representation is extracted through differentiable functions. They also demonstrated that a “natural image prior” is necessary to reconstruct interpretable images. They regularized reconstructed images to eliminate spikes in raw pixels and to be within the natural RGB range. Simonyan et al. [47] adopted a similar approach and used $L_2$ regularization on images.

Dosovitskiy and Brox [11] inverted CNN features by directly learning a CNN, which translates features to images. They demonstrated that colors and rough compositions of the original image can be reconstructed.

Our reconstruction method is based on the work by Mahendran and Vedaldi [38]. Instead of using a hand-crafted natural image prior, we use RNNLM trained on natural images as a regularizer.

2.3. Eye fixation prediction

Modeling visual attention is fundamentally important to efficiently process massive real-world data. Especially, a task to predict eye fixation points of humans has been examined extensively [3].

Bruce and Tsotsos [5] demonstrated that eye fixation points can be predicted using Shannon’s “self-information”. This information-theoretic view has been adopted for many research efforts [3]. Our method also uses a kind of self-information.

Many recent methods are based on supervised training on an eye fixation dataset [25, 27, 31, 32, 36, 55]. Ours is also a trainable flexible model. However, because it is trained in an unsupervised manner, it requires images of the target domain but does not require eye fixation data. Because making a dataset is a troublesome task, it is a favorable property for practical applications.

3. Visual language modeling

As illustrated in Figure 1, we measure the naturalness of an image using RNNLM and CNN. The pipeline of our method and corresponding sections are explained below.

1. Extract a mid-level image representations from an input image using a pre-trained CNN. (Section 3.1.1.)
2. Normalize and orthogonalize them. (Section 3.2.)
3. Run RNNLM forward and backward along the x-axis and y-axis to obtain prediction maps and prediction error maps. (Section 3.1.2, 3.3.)

4. Sum them up and output it as the unnaturalness score. (Section 3.3.)

RNNLM must be trained on natural images in advance. The training procedure is described in Section 3.4.

We apply our method to image reconstruction from features and eye fixation prediction. We describe the details of two applications in Section 3.5 and Section 3.6.

3.1. CNN and RNN

First, we briefly introduce CNN and RNN. We also describe their configuration in this work.

3.1.1 CNN

CNN is a neural network that achieves state-of-the-art performance for image classification [30, 48]. To extract image representations, a CNN applies 2D convolution, nonlinear activation function, and downsampling in a hierarchical way. The weights of convolution kernels are learned from data to minimize classification error.

Mid-level representations of CNNs trained on a large-scale generic image classification dataset are shown to work as a high-performance generic image feature [10, 48]. Therefore they have become the de facto standard image feature in recent years.

We use the outputs immediately after the convolution layers of AlexNet [30] and VGGNet [48] to extract mid-level representations. Table 1 shows layer name in the Caffe pre-trained model [24] and the output size of their convolution layers.

| Model       | Layer name   | Output size          |
|-------------|--------------|----------------------|
| AlexNet [30]| input        | $227 \times 227 \times 3$ |
|             | conv1        | $55 \times 55 \times 96$ |
|             | conv2        | $27 \times 27 \times 256$ |
|             | conv3        | $13 \times 13 \times 384$ |
|             | conv4        | $13 \times 13 \times 384$ |
|             | conv5        | $13 \times 13 \times 256$ |
| VGGNet [48] | conv1\{1,2\} | $224 \times 224 \times 64$ |
|             | conv2\{1,2\} | $112 \times 112 \times 128$ |
|             | conv3\{1,2,3,4\} | $56 \times 56 \times 256$ |
|             | conv4\{1,2,3,4\} | $28 \times 28 \times 512$ |
|             | conv5\{1,2,3,4\} | $14 \times 14 \times 512$ |

Table 1. Layer name and output size of convolution layers. The output size is represented as $height \times width \times dimension$. “conv\{1,2\}” signifies that there are two layers named “conv1” and “conv2”.

4. Sum them up and output it as the unnaturalness score. (Section 3.3.)

RNNLM must be trained on natural images in advance. The training procedure is described in Section 3.4.

We apply our method to image reconstruction from features and eye fixation prediction. We describe the details of two applications in Section 3.5 and Section 3.6.

3.1.2 RNN

RNN is a neural network used to predict a sequence given a sequence. For each timestep $t$, hidden unit $h_t$ receives infor-
mation from input $x_t$ and previous hidden unit $h_{t-1}$. Then $h_t$ passes information to output $y_t$. Because $h_t$ and $h_{t-1}$ are connected, $y_t$ depends on $x_1, x_2, ..., x_t$. Actually, RNN can make predictions using the context of infinite length.

The most basic type of recurrent layer is formalized as follows.

$$h_t = \tanh (W_x x_t + W_h h_{t-1} + b) . \quad (1)$$

However, it cannot learn long-term dependencies in fact because gradients vanish in the process of flowing through many hidden-to-hidden connections. To overcome this problem, a variant of recurrent layer called Long-Short Term Memory (LSTM) was proposed [20]. We used a LSTM recurrent layer defined as shown below.

$$i_t = \sigma (W_{xi} x_t + W_{hi} h_{t-1} + b_i). \quad (2)$$

$$f_t = \sigma (W_{xf} x_t + W_{hf} h_{t-1} + b_f). \quad (3)$$

$$o_t = \sigma (W_{xo} x_t + W_{ho} h_{t-1} + b_o). \quad (4)$$

$$\tilde{c}_t = \tanh (W_{xc} x_t + W_{hc} h_{t-1} + b_c). \quad (5)$$

$$c_t = i_t \odot \tilde{c}_t + f_t \odot c_{t-1}. \quad (6)$$

$$h_t = o_t \odot \tanh (c_t). \quad (7)$$

Therein, $\sigma$ represents a sigmoid activation function.

We use RNN using LSTM for sequential prediction, as described later. Concretely, we stack two LSTM layers and one linear layer to predict sequences. We set the dimensions of two LSTM layers as half of the input layer.

### 3.2. Preprocessing

For example, the representation after conv1 layer of AlexNet comprises 55 × 55 vectors of 96 dimensions. We normalize each dimension of such vectors to have zero-mean and unit-variance. After normalization, we apply Principal Component Analysis (PCA) on it to orthogonalize and reduce dimensions by half. Parameters of normalization and PCA are learned from the training set of ILSVRC 2012 image classification dataset [44].

### 3.3. Combination of CNN and RNNLM

Language Models (LMs) are used to measure the naturalness of a sequence. Let $s_1, s_2, ..., s_T$ to be a sequence of $D$ dimensional vectors of length $T$. The LMs decompose the probability $p(s)$ into $p(s_1)p(s_2|s_1)p(s_3|s_1, s_2)...p(s_T|s_1, s_2, ..., s_{T-1})$. An intuitive interpretation of this is that the probability is determined by how the next vector is predictable from past vectors.

Common LMs treat $s_i$ of one-hot vector which represents a word and assume multinoulli distribution on $p(s_i|s_1, ..., s_{t-1})$. In contrast, we use dense real-value vector $s_t$ and assume Gaussian distribution on $p(s_t|s_1, ..., s_{t-1})$. Concretely, regarding the Gaussian distribution, we assume that the mean $\mu_t$ is determined from $s_1, ..., s_{t-1}$ using RNN described in Section 3.1.2. We also assume that the variance of timestep $t$ is $\frac{1}{T}$ because where $t$ is small the model does not know much “context” of the sequence and predicted values are less reliable. Using this assumption, $p(s_t|s_1, ..., s_{t-1})$ is rewritten by $\mathcal{N}(s_t|\mu_t, \frac{1}{T})$.

The negative log likelihood of $s$ can be written as follows.

$$-\log p(s) \propto \frac{1}{T} \sum_{t=2}^{T} ||s_t - \mu_t||^2. \quad (8)$$

It is the weighted sum of squared prediction error.

We expand this model to 2D mid-level representation of CNN by application of RNNLM forward and backward for each row and column. We apply the same RNNLM in the same axis and direction. Therefore, there are four RNNLMs per layer. Let $f_{y,x,t}$ to be a representation immediately after layer $t$ of size $H_t \times W_t \times D_t$. Then, we define the “unnaturalness map” $u_{y,x,t}(1 \leq y \leq H_t - 1, 1 \leq x \leq W_t - 1)$ of $f_t$ as follows.

$$u_{y,x,t} = \frac{x + 1}{W_t} \left| \left| f_{y,x,t+1} - \mu^\text{right}_{y,x,t} \right| \right|^2 + \frac{W_t - x + 1}{W_t} \left| \left| f_{y,x,t} - \mu^\text{left}_{y,x,t} \right| \right|^2 + \frac{y + 1}{H_t} \left| \left| f_{y+1,x,t} - \mu^\text{down}_{y,x,t} \right| \right|^2 + \frac{H_t - y + 1}{H_t} \left| \left| f_{y,x,t} - \mu^\text{up}_{y,x,t} \right| \right|^2. \quad (9)$$

Therein, $\mu^\text{right}_{y,x,t}$ is predicted value of $f_{y,x,t}$ from $f_{y,1,t}, ..., f_{y,x-1,t}$. This corresponds to scanning of an image from left to right predicting next time-step. $\mu^\text{left}_{y,x,t}$, $\mu^\text{down}_{y,x,t}$, and $\mu^\text{up}_{y,x,t}$ are also defined as the similar way. We define the total unnaturalness of $f_t$ as

$$u_t = \frac{1}{(H_t-1)(W_t-1)} \sum_{y=1}^{H_t-1} \sum_{x=1}^{W_t-1} u_{y,x,t}. \quad (10)$$

We introduce weighting parameter $\lambda_t$ for layer $l$. Then, the total unnaturalness of an image $i$ is

$$u_i = \sum_{l} \lambda_t u_{l,i}. \quad (11)$$

### 3.4. Training of RNNLM

To compute naturalness, we must train RNNLM in advance by minimizing $u_t$ of many natural images. We use Stochastic Gradient Descent (SGD) and Back-Propagation Through Time (BPTT) to train RNNLM. We use the training set of ILSVRC 2012 image classification dataset [44] for training.

We train RNNLM on mid-level representations of AlexNet [30] and VGGNet [48]. For AlexNet, learning rate and momentum of SGD is set, respectively, to 10 and 0.9. The minibatch size is set to 16. For VGGNet, learning rate and momentum of SGD is set, respectively, to 20 and 0.9. The minibatch size is set to 1.

For both networks, we reduce learning rate by the factor of 0.1 for every 5000 iterations. We stop learning after 20,000 iterations.
3.5. Image reconstruction from features

Mahendran and Vedaldi [38] demonstrated that image features can be inverted to the original image by gradient descent (GD) if the feature extraction function comprises differentiable elements. Their key technique is the introduction of “natural image prior” $\mathcal{R}(i)$ into their objective function. We denote the original image as $i$ and feature extraction function as $\phi(i)$. Then, using $\lambda_r$ as the weight of the regularizer, the reconstructed image $\hat{i}$ is

$$\hat{i} = \arg\min_i ||\phi(i) - \phi(\hat{i})||_2^2 + \lambda_r \mathcal{R}(\hat{i}).$$

They used $\mathcal{R}(\hat{i})$, which keeps pixel values in the natural range and penalizes strong intensity change in neighboring pixels. Instead, we set $\mathcal{R}(\hat{i}) = u_i$. Because $u_i$ is differentiable, the objective function can be minimized by GD.

3.6. Eye fixation prediction

It has been suggested that humans look at locations where Shannon’s “self-information” is high [5]. Because self-information is identical to the negative logarithm of probability, unnaturalness map $u_i$ can be regarded as a kind of information map that predicts salient locations.

We use an unnaturalness map $u_i$ as a saliency map. Additionally, we apply Gaussian blur of a size of $\sigma$ to $u_i$ according to common practice [32, 36, 60]. Before blurring, we take the root of $u_i$ to prevent excessive expansion of peaky values. An example of unnaturalness map or saliency map $u_i$ is presented in Figure 2.

4. Experiments

In this section, we present evaluation of the effectiveness of our proposed method by application of it to two tasks: image reconstruction from features and eye fixation prediction.

4.1. Image reconstruction from features

Here we evaluate our image reconstruction method proposed in Section 3.5. First, we discuss how to evaluate reconstructed images. Then we present reconstructed images of ours and compare them with results of existing methods. Subsequently, we combine our method with the work by Dosovitskiy and Brox [11] and present further improved results. Finally, we examine the role of each layer by imposing the regularizer on the target layer.

In common with the preceding works [11, 38], we reconstructed images from the outputs of the last fully-connected layer of AlexNet and used the first one hundred images in the validation set of ILSVRC 2012 classification dataset [44].

Our model has the following hyper-parameters: the set of layers $l$ used for mid-level representations, the weight of unnaturalness map $\lambda_i$, the weight of the regularizer $\lambda_r$, and the learning rate and momentum of GD. In this section, unless otherwise noted, we set $l \in \{\text{conv1}, \text{conv2}, \text{conv3}, \text{conv4}, \text{conv5}\}$, $\lambda_{\text{conv}n} = 10^{-(n-1)}$ for $n \in \{1, 2, 3, 4, 5\}$ and $\lambda_r = 10$. The learning rate and momentum of GD are set, respectively, to $2^{11}$, 0.9. Initial solution of GD is sampled from Gaussian distribution, the mean and standard deviation of which are learned by RGB values of natural images using the training set of ILSVRC 2012 classification dataset [44].

4.1.1 Evaluation method

Quantitative evaluation of whether a reconstructed image is similar to the original image or not is not straightforward. Some earlier reports of the literature [8, 57] have provided no quantitative analysis. Because using a kind of image feature can produce an unfair comparison, mean squared error [11, 28, 38] or correlation coefficient [56] between reconstructed images and original images have been used to date. Vondrick et al. [56] evaluated reconstructed images by asking humans to classify them and reported that the correlation coefficient did not always match the judgments of humans.

Therefore, in this work, we determine similarity of images by asking humans. We provide human subjects with the original image and corresponding reconstructed images. Then they select the image from reconstructed images which is the most similar to the original image. We asked one hundred people on CrowdFlower\footnote{http://www.crowdflower.com/}.

| Method               | Votes | Most similar |
|----------------------|-------|--------------|
| Mahendran and Vedaldi [38] | 1945  | 4            |
| Dosovitskiy and Brox [11]  | 1333  | 2            |
| CNN-VLM (ours)        | 6722  | 94           |
| Total                | 10000 | 100          |

Table 2. Human evaluation of each reconstruction method. For one hundred images, one hundred people on CrowdFlower selected the most similar image to the original image from three reconstructed images. There were a total $100 \times 100 = 10000$ votes. Actually, 94% of our results are selected as the most similar ones to the original images.
Figure 3. Images reconstructed from outputs of the last fully-connected layer of AlexNet.

Figure 5. Images reconstructed from outputs of the last fully-connected layer of AlexNet using our method. Images are initialized with the results of Dosovitskiy and Brox [11].

Figure 4. Six cases for which our results are inferior to other methods according to human evaluation. Top row to bottom row: original image, Mahendran and Vedaldi [38], Dosovitskiy and Brox [11], ours, and ours initialized with the results of Dosovitskiy and Brox. Left four columns: the result of Mahendran and Vedaldi is the best. Right two columns: the result of Dosovitskiy and Brox is the best.

The method presented by Dosovitskiy and Brox [11] reconstructs overall shapes and colors well, although the details are lost because their method outputs an average of possible solutions. Our results appear to be the most similar to the original images. They are clear and understandable, which helps us to interpret what the image features capture.

Table 2 presents results of quantitative evaluation by crowd sourcing. Of 100 images, 94 of our images are selected to the most similar images to the original images. Our results received 67.22% of total votes by 100 people, which clearly indicates the superiority of our method.

Figure 4 presents six cases in which our results were judged to be inferior to two other methods. Our method is not good with reconstruction of the absolute positions and colors of objects. Additionally, our method is vulnerable to “unnatural” images because it is trained on natural images.

4.1.3 Better initialization

The initial solution of GD is known to affect the result strongly, especially in neural networks [13, 50]. In fact, the key to breakthroughs in deep networks resulted from smart initialization of weights [19].

Because results of the method by Dosovitskiy and Brox can be interpreted as the average of possible solutions, they can be good initial solutions. Figure 5 and the last row of Figure 4 portray reconstructed images initialized with the outputs of the method presented by Dosovitskiy and Brox.

These results were improved considerably from previous results. For some images, the absolute positions and colors of objects are corrected, which indicates that the limitations of our method are mostly attributable to the initialization strategy and that they can be compensated by the method presented by Dosovitskiy and Brox.

4.1.4 Analysis of layers

Figure 6 shows images reconstructed using our method. In this case, the regularizer is imposed on one certain layer or raw image pixels.
Figure 6. Images reconstructed with regularization on various feature maps. (a) Original image. (b) No regularization. (c–f) Regularized by the naturalness of (c) the raw pixels, and the output of (d) conv1, (e) conv2, (f) conv3 convolution layers.

The result regularized on raw pixels, which are completely understandable, indicate the importance of modeling the naturalness of high-level features for generating realistic images, not of raw image pixels.

The results regularized on conv2 or conv3 are less clear than the result on conv1, which implies that the information contained by lower layers affects the naturalness of images. Higher layers can regularize more abstract information, but they are insufficient by themselves for generating images.

4.2. Eye fixation prediction

In this section, we evaluate our eye fixation prediction method proposed in Section 3.6. We describe details of the datasets and evaluation metrics. Subsequently, we present the results.

Our model has two hyper-parameters: the set of layers \( l \) used for mid-level representations and the kernel size of Gaussian blur \( \sigma \). We use one convolution layer of VGGNet [48] as \( l \) and set \( \sigma \) to 0.030.

4.2.1 Dataset

Many datasets are used for eye fixation prediction. Herein, we evaluate our method on two popular datasets, called MIT1003 [27] and Toronto [5]. Additionally, we evaluate ours on MIT Saliency Benchmark [6, 26], which consists of two other datasets: MIT300 and CAT2000. The MIT Saliency Benchmark is an online benchmarking service. Evaluation is done in submission.

**MIT1003** MIT1003 Dataset [27] includes 1003 images of natural indoor and outdoor scenes and corresponding eye fixation maps. It includes 779 landscape images and 228 portrait images.

**Toronto** Toronto [5] includes 200 images of outdoor and indoor scenes and corresponding eye fixation maps.

**MIT300** MIT300 [6, 26] includes 300 images of natural indoor and outdoor scenes. Corresponding fixation maps are not provided. Because the protocol to collect this dataset is almost as the same as MIT1003, MIT1003 is useful as a training set.

**CAT2000** CAT2000 [4] is a recently introduced dataset. It includes 2000 images of 20 different image categories including action, affective, art, black & white, cartoon, fractal, indoor, inverted, jumbled, line drawing, low-resolution, noisy, object, outdoor man-made, outdoor natural, pattern, random, satellite, sketch, and social. This dataset is challenging because most categories are not natural scenes.

4.2.2 Evaluation metrics

Eye fixation task can be interpreted as detection task to detect eye fixation point from an image. Therefore, area-under-the-curve score (AUC) of ROC curve is often used for evaluation [5]. However, because humans tend to look around the center of an image, this metric assigns much value to center-biased saliency maps. To overcome this problem, shuffled AUC has been proposed [52, 61]. This metric computes AUC, not on all pixels uniformly, but on center-biased eye fixation points of other images. The shuffled AUC score of centered Gaussian is about 0.5. We use this metric for evaluation.

4.2.3 Results of benchmark dataset

Figure 7 shows the shuffled AUC score obtained using our method on MIT1003 and Toronto dataset. Various \( l \) and \( \sigma \) are tested. The best performing setting is \( l = \text{conv5}_4, \sigma = 0.030 \) for MIT1003 and \( l = \text{conv5}_1, \sigma = 0.010 \) for Toronto.

\(^2\)This value is not posted online but is included in their paper.
Table 3. Shuffled AUC score of each method and dataset. Scores aside from ours are cited from Liu et al. [36].

| Dataset | Mr-CNN | AWS | BMS | CA | eDN | HFT | ICL | IS | JUDD | LG | QDCT | Ours |
|---------|--------|-----|-----|----|-----|-----|-----|----|-------|----|-------|------|
| MIT1003 | .7190  | .6945 | .6939 | .6718 | .6273 | .6526 | .6667 | .6686 | .6631 | .6823 | .6686 |
| Toronto | .7236  | .7184 | .7221 | .6959 | .6292 | .6926 | .6939 | .7115 | .6901 | .6990 | .7174 |
| MIT300  | .7190  | .7190 | .7190 | .6990 | .6990 | .6990 | .6990 | .6990 | .6990 | .6990 | .6990 |
| CAT2000 | .7203  | .7203 | .7203 | .7203 | .7203 | .7203 | .7203 | .7203 | .7203 | .7203 | .7203 |

Table 3 presents our results and existing results on the MIT1003 and Toronto datasets. We compare ours with Mr-CNN [36], AWS [12], BMS [60], CA [14], eDN [55], HFT [35], ICL [22], IS [21], JUDD [27], LG [2] and QDCT [46]. Ours achieved state-of-the-art score on these well-studied datasets.

Table 4. Shuffled AUC score on MIT Saliency Benchmark. Scores are available online. These scores are retrieved on 30 October 2015.

| Dataset | SALICON | Deep Fix | Deep Gaze I² | SalNet | Mr-CNN | AWS | CA | WMAP | IttiKoch2 | Ours |
|---------|---------|----------|-------------|--------|--------|-----|----|------|-----------|------|
| MIT300  | .74     | .71      | .71         | .69    | .69    | .68 | .65 | .63  | .63       | .7096|
| CAT2000 | -       | .57      | -           | -      | -      | -   | .62 | .60  | .59       | .6221|

Table 5. Category-wise results on CAT2000 dataset. Others are averages of all published results available on the scoreboard. Categories for which the difference between ours and others are small or large are shown.

Table 5 presents our score and the average of scores of submitted results available on the scoreboard of each category on the CAT2000 dataset. Compared to other methods, ours are superior for Social, Cartoon, and Affective. Presumably, that is true because images of these categories include more high-level contents such as faces or pedestrians. Ours is inferior for Pattern, Low Resolution, and Fractal because ours are trained on natural images. Images of these categories are apparently not natural. Prediction accuracy can be improved by training of RNNLM on images of these categories.

Most top-scoring methods on MIT Saliency Benchmark are based on supervised training on eye-fixation dataset [25, 31, 32, 36]. These methods require large eye fixation dataset of target domain. If there is no dataset or the dataset is small, the performance of these models can drop. The difference of scores of DeepFix on MIT300 and CAT2000 indicates that. Ours are based on unsupervised training. Therefore it is unaffected by this problem.

5. Conclusion
In this work, based on an assumption that the naturalness can be measured by the predictability on high-level features during eye movement, we proposed a novel method to measure the naturalness of an image by building a variant of RNNLMs on the CNN features. We used it as a regularizer in reconstructing images from image features. The results of experiments show that this regularizer helps to generate more feasible images than existing approaches. We found that the naturalness of lower layers is important to generate natural images. Additionally, we evaluated “unnaturalness maps” of images as saliency maps. This was motivated by the assumption that saliency of images is based on the self-information of each location. We demonstrated that the proposed “unnaturalness map” achieves state-of-the-art shuffled AUC scores on two well-studied eye fixation prediction datasets. It was indicated that the naturalness of higher layers predicts eye fixation points well.

The naturalness of images, especially for large images that include rich contents, has not been studied well. Nonetheless, methods to assess and detect naturalness of images are extremely useful, as demonstrated in the experiments. We hope this work will provoke more active research in this field.

3Unpublished work.
References

[1] Y. Bengio, E. Thibodeau-Laufer, G. Alain, and J. Yosinski. Deep generative stochastic networks trainable by backprop. In ICML, 2014. 1, 2

[2] A. Borji and L. Itti. Exploiting local and global patch rarities for saliency detection. In CVPR, 2012. 8

[3] A. Borji and L. Itti. State-of-the-art in visual attention modeling. PAMI, 35(1):185–207, 2013. 2, 3

[4] A. Borji and L. Itti. Cat2000: A large scale fixation dataset for boosting saliency research. In CVPR Workshop on the Future of Datasets in Vision, 2015. 7

[5] N. Bruce and J. Tsotsos. Saliency based on information maximization. In NIPS, 2005. 2, 3, 5, 7

[6] Z. Bylinskii, T. Judd, F. Durand, A. Oliva, and A. Torralba. Mit saliency benchmark, 2012. http://saliency.mit.edu. 7

[7] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In ECCV Workshop on Statistical Learning in Computer Vision, 2004. 2

[8] E. d’Angelo, A. Alahi, and P. Vandergheynst. Beyond bits: Reconstructing images from local binary descriptors. In ICPR, 2012. 3, 5

[9] E. Denton, S. Chintala, A. Szlam, and R. Fergus. Deep generative image models using a laplacian pyramid of adversarial networks. In NIPS, 2015. 2

[10] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In ICML, 2014. 1, 3

[11] A. Dosovitskiy and T. Brox. Inverting convolutional networks with convolutional networks. arXiv:1506.02753, 2015. 3, 5, 6

[12] A. Garcia-Diaz, V. Leborán, X. R. Fdez-Vidal, and X. M. Pardo. On the relationship between optical variability, visual saliency, and eye fixations: A computational approach. Journal of Vision, 12(6):17, 2012. 8

[13] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In AISTATS, 2010. 6

[14] S. Goferman, L. Zelnik-Manor, and A. Tal. Context-aware saliency detection. PAMI, 34(10):1915–1926, 2012. 8

[15] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In NIPS, 2014. 2

[16] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. In ICLR, 2015. 1

[17] A. Graves and J. Schmidhuber. Offline handwriting recognition with multidimensional recurrent neural networks. In NIPS, 2009. 2

[18] K. Gregor, I. Danihelka, A. Graves, and D. Wierstra. Draw: A recurrent neural network for image generation. In ICML, 2015. 2

[19] G. E. Hinton, S. Osindero, and Y.-W. Teh. A fast learning algorithm for deep belief nets. Neural computation, 18(7):1527–1554, 2006. 1, 2, 6

[20] S. Hochreiter and J. Schmidhuber. Long short-term memory. Neural computation, 9(8):1735–1780, 1997. 4

[21] X. Hou, J. Harel, and C. Koch. Image signature: Highlighting sparse salient regions. PAMI, 34(1):194–201, 2012. 8

[22] X. Hou and L. Zhang. Dynamic visual attention: Searching for coding length increments. In NIPS, 2009. 8

[23] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. PAMI, (11):1254–1259, 1998. 8

[24] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In ACMMM, 2014. 3

[25] M. Jiang, S. Huang, J. Duan, and Q. Zhao. Salicon: Saliency in context. In CVPR, 2015. 3, 8

[26] T. Judd, F. Durand, and A. Torralba. A benchmark of computational models of saliency to predict human fixations. In MIT technical report, 2012. 7

[27] T. Judd, K. Ehinger, F. Durand, and A. Torralba. Learning to predict where humans look. In ICCV, 2009. 3, 7, 8

[28] H. Kato and T. Harada. Image reconstruction from bag-of-visual-words. In CVPR, 2014. 3, 5

[29] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In ICLR, 2014. 1, 2

[30] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In NIPS, 2012. 3, 4

[31] S. S. Kruthiventi, K. Ayush, and R. V. Babu. Deepfix: A fully convolutional neural network for predicting human eye fixations. arXiv:1510.02927, 2015. 3, 8

[32] M. Kümmerer, L. Theis, and M. Bethge. Deep gaze i: Boosting saliency prediction with feature maps trained on imagenet. In ICLR Workshop, 2014. 3, 5, 8

[33] H. Larochelle and I. Murray. The neural autoregressive distribution estimator. In AISTATS, 2011. 1, 2

[34] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998. 1

[35] J. Li, M. D. Levine, X. An, X. Xu, and H. He. Visual saliency based on scale-space analysis in the frequency domain. PAMI, 35(4):996–1010, 2013. 8

[36] N. Liu, J. Han, D. Zhang, S. Wen, and T. Liu. Predicting eye fixations using convolutional neural networks. In CVPR, 2015. 3, 5, 8

[37] F. López-García, R. Doslí, X. M. Pardo, and X. R. Fdez-Vidal. Scene Recognition through Visual Attention and Image Features: A Comparison between SIFT and SURF Approaches. INTECH Open Access Publisher, 2011. 8

[38] A. Mahendran and A. Vedaldi. Understanding deep image representations by inverting them. In CVPR, 2015. 2, 3, 5, 6

[39] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký, and S. Khudanpur. Decaf: A deep convolutional architecture for fast feature embedding. In CVPR, 2015. 2

[40] A. Mordvintsev, C. Olah, and M. Tyka. Inceptionism: Going deeper into neural networks. Google Research Blog Retrieved October 24, 2015. 1

[41] A. Nguyen, J. Yosinski, and J. Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In CVPR, 2015. 1
[42] M. Ranzato, A. Szlam, J. Bruna, M. Mathieu, R. Collobert, and S. Chopra. Video (language) modeling: a baseline for generative models of natural videos. *arXiv:1412.6604*, 2014.

[43] R. Rosenfeld. Two decades of statistical language modeling: Where do we go from here. In *Proceedings of the IEEE*, 2000.

[44] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge. *IJCV*, pages 1–42, 2015.

[45] R. Salakhutdinov and G. E. Hinton. Deep boltzmann machines. In *AISTATS*, 2009.

[46] B. Schauerte and R. Stiefelhagen. Quaternion-based spectral saliency detection for eye fixation prediction. In *ECCV*, 2012.

[47] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *ICLR Workshop*, 2014.

[48] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[49] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, 2003.

[50] I. Sutskever, J. Martens, G. Dahl, and G. Hinton. On the importance of initialization and momentum in deep learning. In *ICML*, 2013.

[51] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. In *ICLR*, 2014.

[52] B. W. Tatler, R. J. Baddeley, and I. D. Gilchrist. Visual correlates of fixation selection: Effects of scale and time. *Vision Research*, 45(5):643–659, 2005.

[53] L. Theis and M. Bethge. Generative image modeling using spatial lstms. In *NIPS*, 2015.

[54] P. Tirilly, V. Claveau, and P. Gros. Language modeling for bag-of-visual words image categorization. In *ICCV*, 2008.

[55] E. Vig, M. Dorr, and D. Cox. Large-scale optimization of hierarchical features for saliency prediction in natural images. In *CVPR*, 2014.

[56] C. Vondrick, A. Khosla, T. Malisiewicz, and A. Torralba. Hoggles: Visualizing object detection features. In *ICCV*, 2013.

[57] P. Weinzaepfel, H. Jégou, and P. Pérez. Reconstructing an image from its local descriptors. In *CVPR*, 2011.

[58] L. Wu, M. Li, Z. Li, W.-Y. Ma, and N. Yu. Visual language modeling for image classification. In *ACMMM Workshop on Multimedia Information Retrieval*, 2007.

[59] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014.

[60] J. Zhang and S. Sclaroff. Saliency detection: A boolean map approach. In *ICCV*, 2013.

[61] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cotrell. Sun: A bayesian framework for saliency using natural statistics. *Journal of Vision*, 8(7):32, 2008.