Generating Class-conditional Images with Gradient-based Inference

Bowen Xu
University of Toronto
xubo3@cs.toronto.edu

David Acuña
University of Toronto
davidj@cs.toronto.edu

David Duvenaud
University of Toronto
duvenaud@cs.toronto.edu

Abstract

Gradient descent on images is a promising approach to generating high-resolution class-conditional images with crisp details and coherent overall structure. Previous work synthesized images by maximizing $p(\text{class}|\text{image})p(\text{image})$, with pre-trained models specifying these two terms. However, this maximization often produces unrepresentative super-stimuli. Instead, we sample from $p(\text{image}|\text{class})$ using gradient-based MCMC methods. This approach produces realistic and content-diverse class-conditional images, and removes the need for *ad-hoc* tweaks to the objective when longer iterations are introduced.

1 Introduction

Current cutting-edge image generation techniques are generally based on Generative Adversarial Networks (GAN) [1] or Variational Autoencoders (VAE) [2]. The Deep Convolutional Generative Adversarial Networks (DCGAN) [3] is the well-known state-of-the-art technique in this field and the DRAW Algorithm [4] is the cutting-edge extension of the VAE. Although DCGAN and DRAW can produce promising low-resolution images on the CIFAR and MNIST datasets [3; 4], they struggle to produce high-resolution images on the ImageNet and MS COCO datasets [5; 6].

A completely different approach, due to Nguyen et. al. [7], uses gradient-based optimization of images to do approximate *maximum a posteriori probability* (MAP) estimation, synthesizing large (227 x 227) class-conditional images. An initial image is specified by a random vector $z \sim \mathcal{N}(0, I)$ which is passed through a pre-trained image generation network $g_\theta(z)$. This image is fed into a pre-trained classification network to compute class probabilities. The latent vector is then optimized according to the gradient of the chosen class probability with respect to the image, multiplied by the gradient of the image with respect to the latent vector, using backpropagation.

Image optimization is more expensive than most class-conditional image generation techniques, but is able to produce large, high-resolution images with both crisp details and coherent overall structure.

Method	Interpolate within Class	Details Crisp	Coherent Overall Structure	High Resolution	Fast Generation
Class-Conditional GAN	✓	✓	×	×	✓
Image optimization	×	✓	✓	✓	×

Table 1: Comparison between different generative methods

The experiments of [7], compose two types of pre-trained neural networks together. First, an adversarially-trained image generation network from [8], trained on ImageNet [9]. This network is used as the image prior $p(\text{image})$ that takes in an input vector and outputs a synthetic image. Second, off-the shelf image classification networks (CaffeNet [10], AlexNet [11], or GoogleNet [12]) are used as likelihoods $p(\text{class}|\text{image})$.

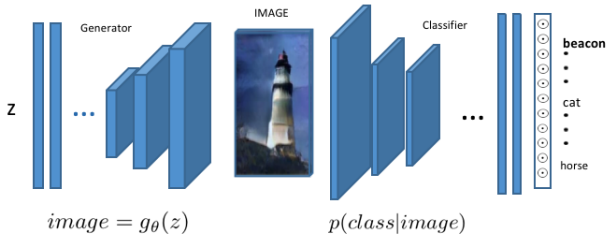

Figure 1: Architecture using the DGN and DNN.

In the experiments of [7], the input vector z is optimized to maximize a regularized loss. We note that this loss is equivalent to the conditional log-probability $\log p(g_\theta(z)|\text{class})$:

$$\hat{z} = \arg\max_z \phi(g_\theta(z)) - \lambda\|z\|_2^2 \quad (1)$$
$$= \arg\max_z \log p(class|g_\theta(z)) + \log p(z) \quad (2)$$
$$= \arg\max_z \log p(g_\theta(z)|\text{class}) \quad (3)$$

Although maximizing (1) can generate high resolution class-conditional images, we observe that this method tends to produce over-saturated super-stimuli unless the latent vectors z are bounded within class-specific regions based on empirical statistic for that class. This *ad-hoc* fix is referred to in [7] as *clipping*. One of our findings is that clipping is unnecessary if the method is run for several thousand iterations, instead of 200 iterations as in [7]. Figure 2 compares images generated using [7]'s approach both with and without clipping, with and without extra iterations, and using gradient-based MCMC methods.

2 Sampling-based Conditional Image Generation

To produce class-conditional images that are more representative of the true distribution of images containing that class, we adopt the Hamiltonian Monte Carlo (HMC) [13] and the Metropolis-adjusted-Langevin-algorithm (MALA) [14] to sample images from $p(\text{image}|\text{class})$.

Markov Chain Monte Carlo (MCMC) can be used to sample from complex distributions. However, the non-adaptive proposal distributions of many Metropolis-Hastings methods can lead to slow mixing of the Markov chain. To address such problems, some MCMC methods use the gradient information of the log-posterior to guide the sampler towards high-density regions. One such method is called the Metropolis-adjusted-Langevin-algorithm (MALA) [14] which uses a combination of Langevin diffusion and Metropolis-Hasting acceptance criteria to propose the new states of the random walk. Another method called the Hamiltonian Monte Carlo (HMC) [13] incorporates the Hamiltonian dynamics into the MCMC method by introducing an auxiliary variable, the momentum, into the process.

We use gradient-based MCMC to approximately sample from the class-conditional posterior:

$$\hat{z} \sim p(z|\text{class}) \propto p(\text{class}|g_\theta(z))p(z) \qquad \text{where} \qquad p(z) = \mathcal{N}(0, I) \quad (4)$$

and $p(\text{class}|g_\theta(z))$ is defined by a classifier neural network.

We compare MAP and MALA image generation algorithms below. Note that in [7], the input vector z is bounded through clipping (line 5 of Algorithm 1). On the other hand, MALA uses the Metropolis-Hasting acceptance criterion (line 5 of Algorithm 2). Similarly, HMC also has an Metropolis-Hasting accept/reject step; however, it uses Hamiltonian dynamics for the proposal distribution [13].

Algorithm 1 MAP with Clipping[7]

1: $z \sim N(0, I)$
2: **for** iterations **do**
3: $\quad g = \nabla log(p(\text{class}|z)p(z))$
4: $\quad z = z + \alpha g$
5: $\quad z = clip(z)$
6: **end for**

Algorithm 2 MALA

1: $z \sim N(0, I)$
2: **for** iterations **do**
3: $\quad g = \nabla log(p(\text{class}|z)p(z))$
4: $\quad \hat{z} = z + \alpha g + \sqrt{2\alpha}\epsilon$
5: $\quad z = \text{MH Accept/Reject}(z, \hat{z})$
6: **end for**

3 Experiments and Results

In the following subsections, we first present the advantages of using MCMC methods to synthesize images. Then, we measure how natural the images are by feeding them into the discriminator of a DCGAN [3].

In our experiments, we use [7]'s image generator and classifiers (AlexNet and CaffeNet). Our MALA incorporates annealed Gaussian noise [15], both MALA and HMC use decaying step size schedules [16], and our combined iteration and leapfrog cost is 6000 runs.

We also examine the effect of using the clipping mechanism on MALA. In addition to the standard MALA, we have a modified version where clipping is added after the Metropolis-Hasting acceptance criterion.

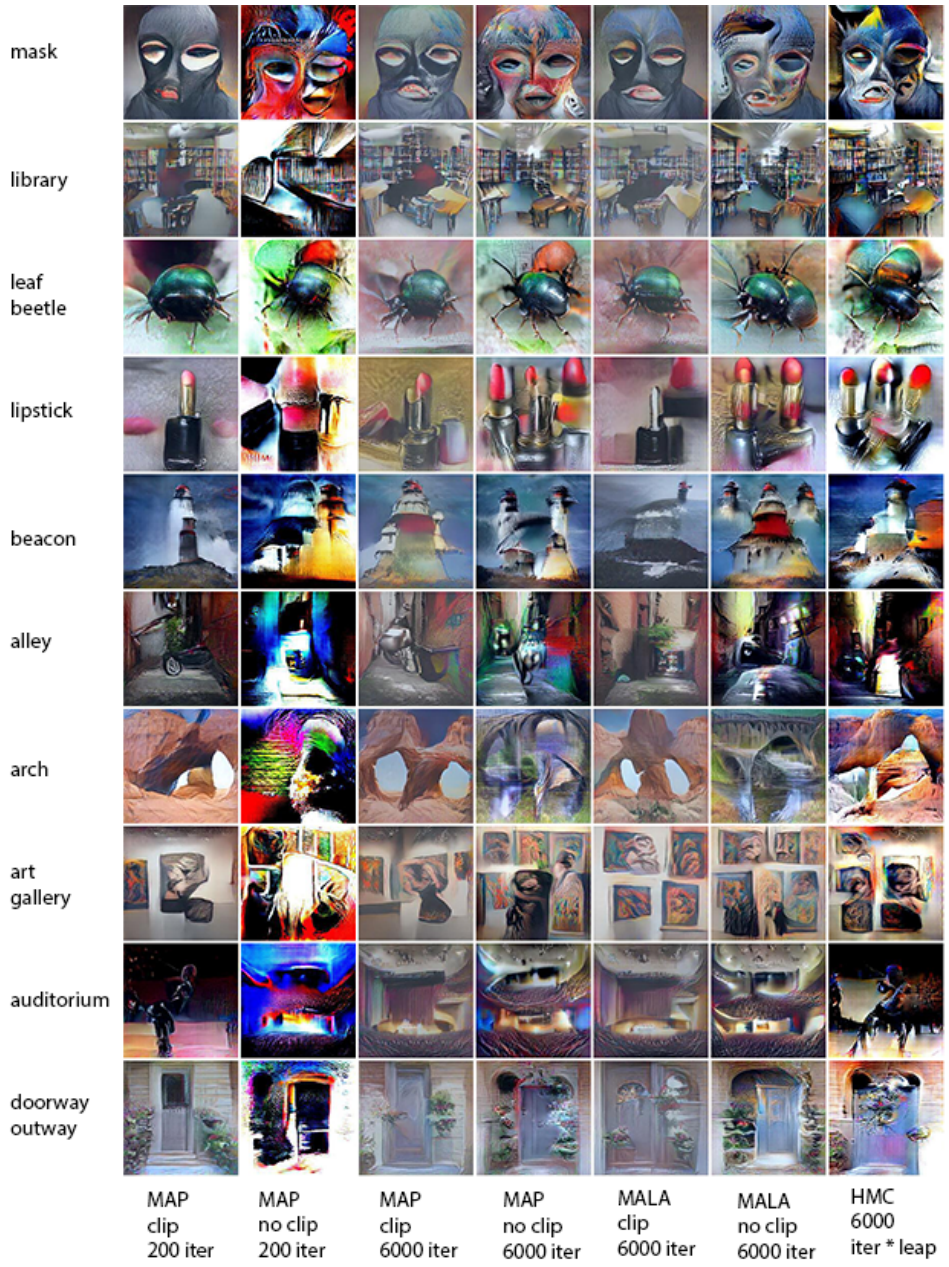

Figure 2: Images generated with MAP, MALA, and HMC.

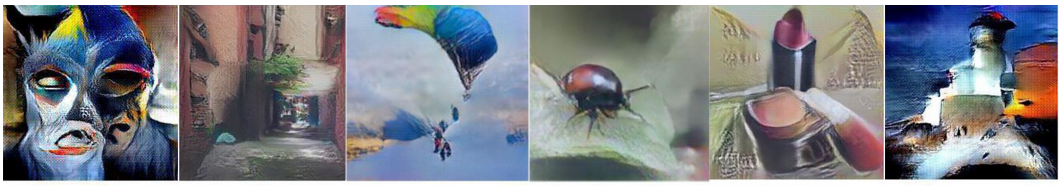

Figure 3: Content-diverse images generated with MALA and HMC

3.1 Image Samples

Figure 2 shows images sampled with MALA and HMC. In contrast to MAP inference, we observe that the MCMC methods produce realistic images even without the clipping mechanism. The sampled images not only correctly illustrate their corresponding classes but also look natural. We also observe that combining the clipping mechanism with MALA reduces the required number of iterations substantially.

3.2 Diversity of Image Contents

We notice also that given enough iterations, gradient-based MCMC methods can generate content-diverse images. For example, Figure 3 shows a picture of a beacon that notably also includes a sunset, as well as an image of lipstick that also contains other types of makeup.

Note that in Figure 3, images for the parachute, alley, leaf beetle, and lipstick classes are generated using MALA with clipping.

3.3 Automatically Evaluating Image Quality

In order to determine how natural the images are, we feed the synthesized images into the discriminator of [3]'s DCGAN. The discriminator, trained on ImageNet, outputs the probability of an image being natural versus synthesized from a GAN. Table 2 shows that the average log-probabilities of gradient-based MCMC methods are very close to that of the MAP method. In addition, when MALA is combined with clipping, the discriminator has a stronger belief that the image is real.

	MAP with Clipping[7]	MALA with Clipping	MALA	HMC
Average log-probability	-6.00	**-5.53**	-9.79	-7.96

Table 2: Average log-probability of synthetic images being real.

4 Limitations

MCMC methods are known for slow convergence, particularly in high-dimensional data spaces [13]. Small step sizes and high number of iterations are often adopted to help improving the mixing rate. However, this is computationally expensive and time consuming. In our experiments, we use an iteration size of 6000, whereas the approach presented in [7] only requires 200 iterations.

5 Conclusions

We have shown that MALA and HMC are robust mechanisms that do not require the *ad-hoc* clipping constraints to generate realistic looking images from $p(\text{image}|\text{class})$. We have also shown that MALA and HMC are able to generate more content-diverse images that display not only the object of the class but also objects that are closely associated to the class. In addition, we have shown that clipping can be combined with gradient-based MCMC methods to improve the quality of the images.

References

[1] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Networks," *ArXiv e-prints*, June 2014.

[2] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," *ArXiv e-prints*, Dec. 2013.

[3] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," *arXiv*, pp. 1–15, 2015.

[4] K. Gregor, I. Danihelka, A. Graves, D. Jimenez Rezende, and D. Wierstra, "DRAW: A Recurrent Neural Network For Image Generation," *ArXiv e-prints*, Feb. 2015.

[5] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative Adversarial Text to Image Synthesis," *ArXiv e-prints*, May 2016.

[6] E. Mansimov, E. Parisotto, J. Lei Ba, and R. Salakhutdinov, "Generating Images from Captions with Attention," *ArXiv e-prints*, Nov. 2015.

[7] A. Nguyen, A. Dosovitskiy, J. Yosinski, T. Brox, and J. Clune, "Synthesizing the preferred inputs for neurons in neural networks via deep generator networks," *arXiv*, pp. 1–29, 2016.

[8] A. Dosovitskiy and T. Brox, "Generating images with perceptual similarity metrics based on deep networks," *CoRR*, vol. abs/1602.02644, 2016.

[9] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, pp. 211–252, 12 2015.

[10] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional Architecture for Fast Feature Embedding," *ACM International Conference on Multimedia*, pp. 675–678, 2014.

[11] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning Deep Features for Scene Recognition using Places Database," *Advances in Neural Information Processing Systems 27*, pp. 487–495, 2014.

[12] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[13] R. M. Neal, "MCMC using Hamiltonian dynamics," *Handbook of Markov Chain Monte Carlo*, vol. 54, pp. 113–162, 2010.

[14] M. Girolami, B. Calderhead, and S. A. Chin, "Riemann manifold langevin and hamiltonian monte carlo methods," *J. of the Royal Statistical Society, Series B (Methodological*.

[15] A. Neelakantan, L. Vilnis, Q. V. Le, I. Sutskever, L. Kaiser, K. Kurach, and J. Martens, "Adding Gradient Noise Improves Learning for Very Deep Networks," *ArXiv e-prints*, Nov. 2015.

[16] Y. Whye Teh, A. Thiéry, and S. Vollmer, "Consistency and fluctuations for stochastic gradient Langevin dynamics," *ArXiv e-prints*, Sept. 2014.