Melhorando o Realismo de Imagens Sintéticas

Vol. 1, edição 1 · julho de 2017

https://machinelearning.apple.com/2017/07/07/GAN.html

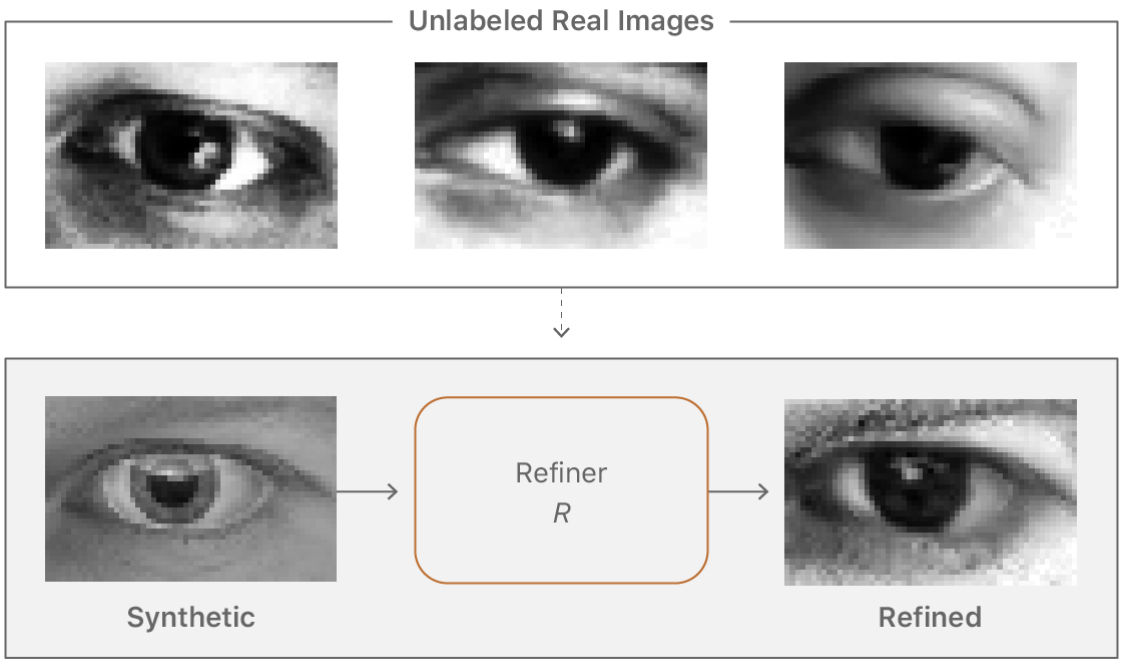
Os exemplos mais bem sucedidos de redes neurais hoje são treinados com supervisão. No entanto, para alcançar uma alta precisão, os conjuntos de treinamento precisam ser largos, variados e com exatidão, o que é dispendioso. Uma alternativa para rotular enormes quantidades de dados é usar imagens sintéticas a partir de um simulador. Isso é barato, pois não há custo de rotulagem, mas as imagens sintéticas podem não ser suficientemente realistas, resultando em uma generalização fraca em imagens de teste reais. Para ajudar a fechar esta lacuna de desempenho, desenvolvemos um método para refinar imagens sintéticas para que elas pareçam mais realistas. Mostramos que os modelos de treinamento nessas imagens refinadas levam a melhorias significativas na precisão em várias tarefas de aprendizagem de máquinas.

Visão geral

A formação de modelos de aprendizagem de máquinas em imagens sintéticas padrão é problemática, pois as imagens podem não ser suficientemente realistas, levando o modelo a aprender detalhes presentes apenas em imagens sintéticas e não generalizando bem em imagens reais. Uma abordagem para colmatar essa lacuna entre imagens sintéticas e reais seria melhorar o simulador, que muitas vezes é caro e difícil, e mesmo o melhor algoritmo de renderização ainda pode deixar de modelar todos os detalhes presentes nas imagens reais. Essa falta de realismo pode fazer com que os modelos se supertem aos detalhes "irrealistas" nas imagens sintéticas. Em vez de modelar todos os detalhes no simulador, podemos aprendê-los a partir de dados? Para este fim, desenvolvemos um método para refinar imagens sintéticas para torná-las mais realistas (Figura 1).

Figura 1. A tarefa é aprender um modelo que melhore o realismo de imagens sintéticas a partir de um simulador usando dados reais não

marcados, preservando as informações de anotação.



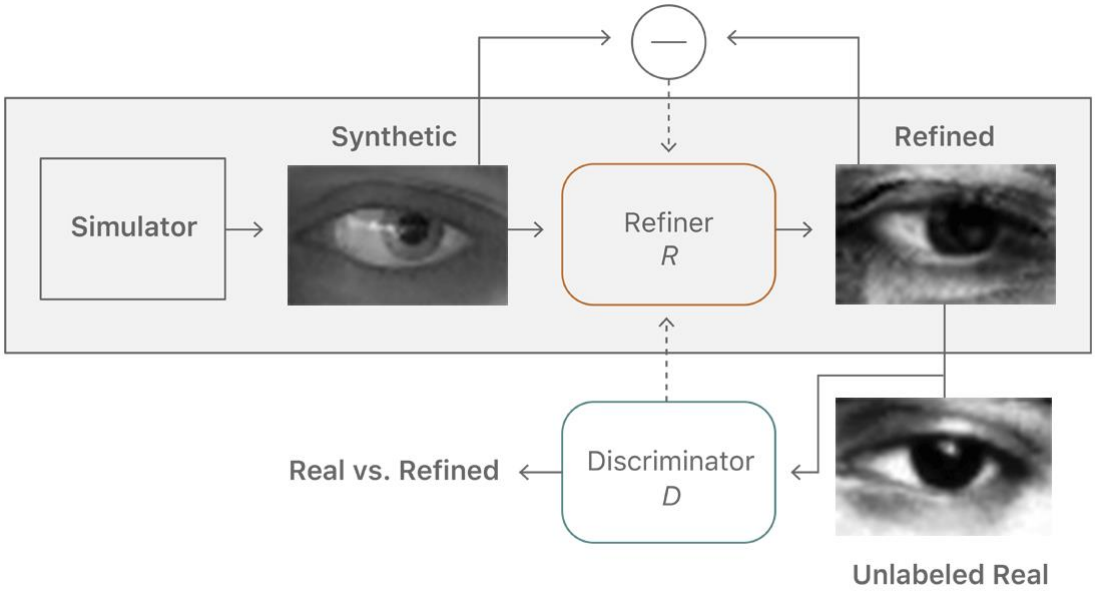
O objetivo de "melhorar o realismo" é fazer com que as imagens pareçam tão realistas quanto possível para melhorar a precisão do teste. Isso significa que queremos preservar informações de anotação para treinamento de modelos de aprendizagem de máquinas. Por exemplo, a direção do olhar na Figura 1 deve ser preservada, além de não gerar nenhum artefato, pois os modelos de aprendizado de máquina podem aprender a superar-se. Aprendemos uma rede neural profunda, que chamamos de "Rede de refinadores", que processa imagens sintéticas para melhorar seu realismo.

Para aprender uma rede de refinadores, precisamos de algumas imagens reais. Uma opção seria exigir pares de imagens reais e sintéticas com correspondência em pixels, ou imagens reais com anotações - por exemplo, a informação do olhar no caso dos olhos. Este é indiscutivelmente um problema mais fácil, mas esses dados são muito difíceis de colecionar. Para criar uma correspondência em pixel, precisamos renderizar uma imagem sintética que corresponde a uma determinada imagem real ou capturar uma imagem real que corresponda a uma imagem sintética renderizada. Podemos, em vez disso, aprender esse mapeamento sem correspondência em pixels ou qualquer rótulo para as imagens reais? Se assim for, podemos gerar um monte de imagens sintéticas, capturar imagens reais de olhos e, sem rotular qualquer imagem real, aprenda este mapeamento, tornando o método barato e fácil de aplicar na prática.

Para aprender nossa rede de refinadores de forma não supervisionada, utilizamos uma rede auxiliar de discriminação que classifica as imagens

reais e refinadas (ou falsas) em duas classes. A rede de refinadores tenta enganar essa rede discriminadora para pensar que as imagens refinadas são as reais. As duas redes se alternam, e o treinamento pára quando o discriminador não consegue distinguir as imagens reais das falsas. A idéia de usar uma rede discriminadora adversária é semelhante à abordagem GANs (Generative Adversarial Networks [1]) que mapeia um vetor aleatório para uma imagem, de modo que a imagem gerada seja indistinguível dos reais. Nosso objetivo é treinar uma rede de refinadores - um gerador - que mapeia uma imagem sintética para uma imagem realista. A Figura 2 mostra uma visão geral do método.

Figura 2. Nossa rede neural de refinador, R, minimiza uma combinação de perda adversarial local e um termo de "auto-regularização". A perda contraditória "engana" a rede discriminadora, D, que classifica uma imagem como real ou refinada. O termo de auto-regularização minimiza a diferença de imagem entre as imagens sintéticas e refinadas. A rede de refinadores e a rede discriminadora são atualizadas alternadamente.



Como preservamos as anotações?

Além de gerar imagens realistas, a rede do refinador deve preservar as informações de anotação do simulador. Por exemplo, para a estimativa do olhar, a transformação aprendida não deve mudar a direção do olhar. Esta restrição é um ingrediente essencial para habilitar o treinamento de um modelo de aprendizagem de máquina que usa as imagens refinadas com as anotações do simulador. Para preservar as anotações de imagens sintéticas, complementamos a perda contraditória com uma perda de auto-regularização L1 que penaliza grandes mudanças entre as imagens sintéticas e refinadas.

Como podemos evitar artefatos?

Fazendo mudanças locais

Outro requisito importante para a rede de refinadores é que ele deve aprender a modelar as características reais da imagem sem apresentar nenhum artefato. Quando treinamos uma única rede de discriminadores fortes, a rede de refinadores tende a enfatizar excessivamente certos recursos de imagem para enganar a atual rede discriminadora, levando a derivar e produzir artefatos. Uma observação chave é que qualquer patch local amostrado da imagem refinada deve ter estatísticas semelhantes a um patch de imagem real. Portanto, ao invés de definir uma rede discriminadora global, podemos definir uma rede discriminadora que classifique separadamente todos os patches de imagem locais (Figura 3). Esta divisão não só limita o campo receptivo e, portanto, a capacidade da rede discriminadora, mas também fornece muitas amostras por imagem para aprender a rede discriminadora. A rede de refinadores também é melhorada por ter múltiplos valores de "perda de realismo" por imagem.

Figura 3. Ilustração da perda adversarial local. A rede discriminadora produz um mapa de probabilidade de aw × h. A função de perda contraditória é a soma das perdas de entropia cruzada sobre os remendos locais.

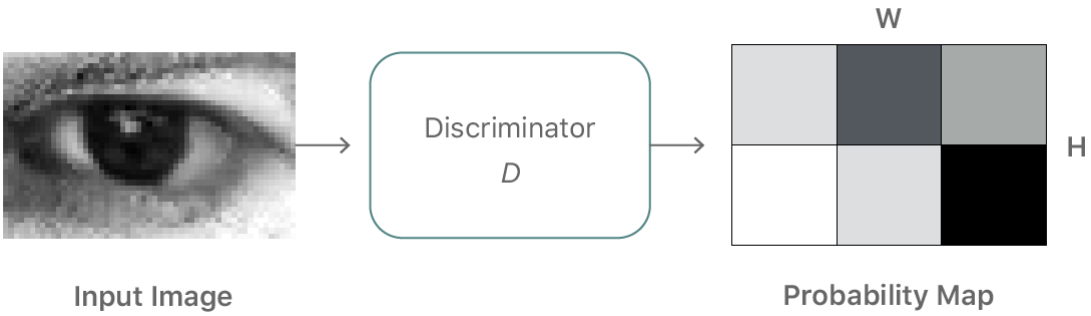


Figura 3. Ilustração da perda adversarial local. A rede discriminadora produz um mapa de probabilidade de largura por altura. A função de perda contraditória é a soma das perdas de entropia cruzada sobre os remendos locais.

Usando o Histórico do Gerador para Melhorar o Discriminador

O gerador pode enganar o discriminador, quer com amostras de uma distribuição nova, quer com a distribuição alvo (dados reais). Gerar a partir de uma nova distribuição engana apenas o discriminador até que o discriminador aprenda essa nova distribuição. A maneira mais útil de que o gerador pode enganar o discriminador é gerando a partir da distribuição de destino.

Dadas estas duas formas de evoluir, o mais fácil é geralmente gerar um novo resultado, que é o que observamos ao treinar o gerador e o discriminador atuais uns contra os outros. Uma ilustração simplificada desta seqüência improdutiva é mostrada no lado esquerdo da Figura 4. As distribuições do gerador e discriminador são mostradas em amarelo e azul, respectivamente.

Ao introduzir uma história que armazena amostras geradoras de gerações anteriores (à direita da Figura 4), o discriminador é menos propenso a esquecer a parte do espaço sobre a qual já aprendeu. O discriminador mais poderoso ajuda o gerador a se mover para a distribuição de destino mais rapidamente. A ilustração é uma simplificação e negligencia mostrar que as distribuições são regiões complexas e muitas vezes desconectadas. Na prática, no entanto, um simples buffer de substituição aleatória capta diversidade suficiente das distribuições anteriores do gerador para evitar a repetição ao fortalecer o discriminador. Nossa noção é que qualquer imagem refinada gerada pela rede de refinadores a qualquer momento durante todo o procedimento de treinamento é realmente uma imagem "falsa" para o discriminador. Descobrimos que, construindo um mini-lote para D com a metade das amostras do buffer de histórico e a outra metade da saída do gerador atual (como mostrado na Figura 5), podemos melhorar o treinamento.

Figura 4. Uma ilustração da intuição por trás do uso de um histórico de imagens para melhorar o discriminador.

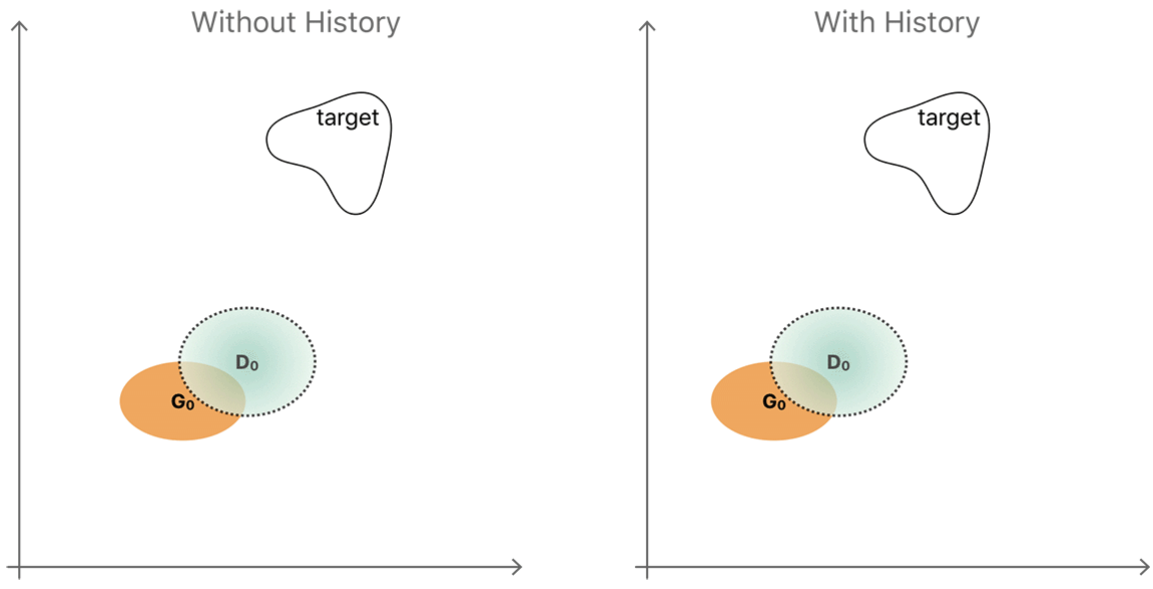
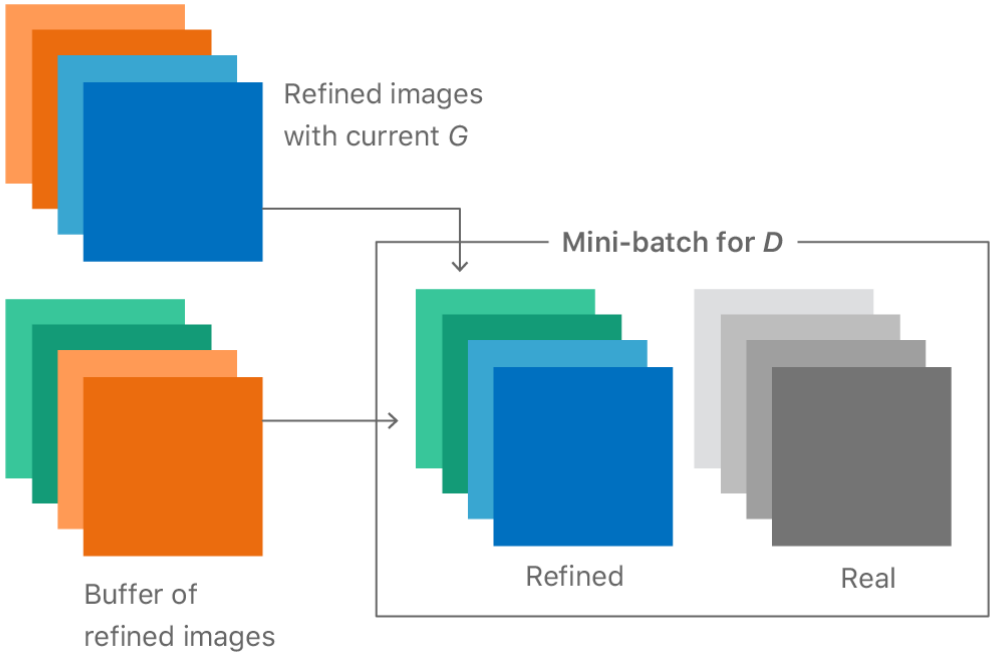


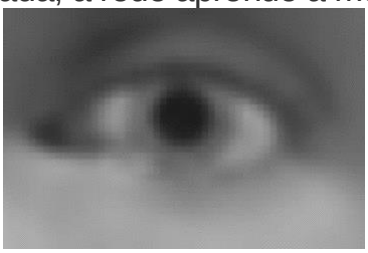
Figura 5. Ilustração do mini-lote com história para D. Cada mini-lote consiste de imagens da iteração atual do gerador, bem como de um buffer de

imagens falsas anteriores.

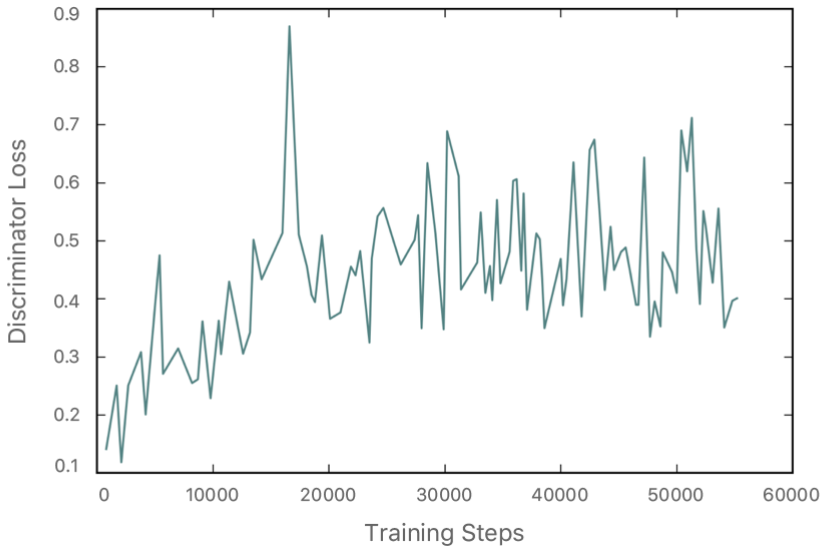


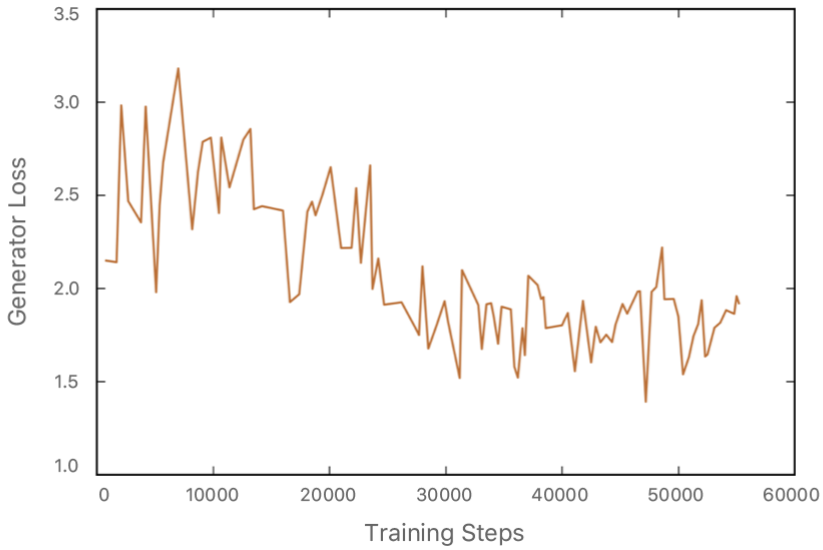
Como o treinamento avança?

Primeiro treinamos a rede de refinadores com apenas perda de auto-regularização e apresentamos a perda adversa depois que a rede de refinadores começa a produzir versões embaçadas das imagens sintéticas de entrada. A Figura 6 mostra a saída da rede de refinadores em várias etapas de treinamento. No início, ele produz uma imagem embaçada que se torna cada vez mais realista à medida que o treinamento avança. A Figura 7 visualiza as perdas do discriminador e do gerador em diferentes iterações de treinamento. Observe que a perda do discriminador é baixa no começo - o que significa que pode facilmente dizer a diferença entre real e refinado. Lentamente, a perda do discriminador aumenta e a perda do gerador diminui à medida que o treinamento avança, gerando mais imagens reais.

Figura 6. Saída da rede de refinadores à medida que o treinamento avança. Começando com uma imagem embaçada, a rede aprende a modelar os detalhes presentes em imagens reais.  Figura 7.

Perdas de gerador e discriminador à medida que o treinamento avança.

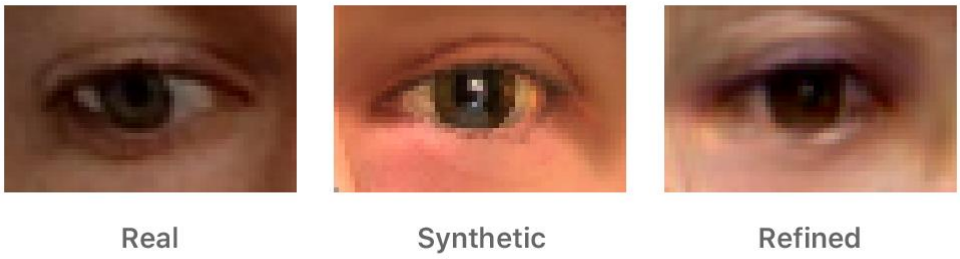




A limitação de perda de L-auto-regularização é limitada?

Quando as imagens sintéticas e reais têm uma mudança significativa na distribuição, uma diferença de L1 em pixels pode ser restritiva. Nesses casos, podemos substituir o mapa de identidade por uma transformação de característica alternativa, colocando um auto-regularizador em um espaço de recursos. Estes poderiam ser recursos ajustados à mão, ou recursos aprendidos, como uma camada intermediária da VGGnet. Por exemplo, para o refinamento da imagem colorida, a média dos canais RGB pode gerar imagens de cores realistas, como a Figura 8.

Figura 8. Exemplo de perda de auto-regularização no espaço de recursos.

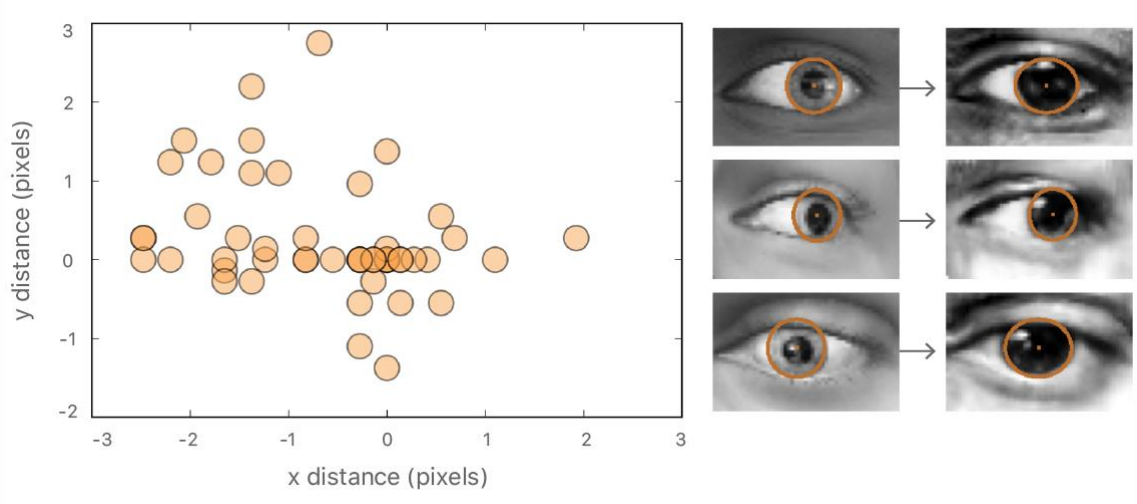


Os rótulos são alterados pelo gerador?

Para verificar se os rótulos não mudam de forma significativa, nós desenhamos as elipses das imagens sintéticas e refinadas, e calcularam a diferença entre seus centros. Na Figura 9, mostramos um gráfico de dispersão de 50 dessas diferenças de centro. A diferença absoluta entre o centro de pupila estimado da imagem sintética e correspondente refinada é bastante pequena: 1.1 +/- 0.8px (largura do olho = 55px).

Para verificar se os rótulos não mudam de forma significativa, nós desenhamos as elipses das imagens sintéticas e refinadas, e calcularam a diferença entre seus centros. Na Figura 9, mostramos um gráfico de dispersão de cinquenta dessas diferenças de centro. A diferença absoluta entre o centro de pupila estimado da imagem refinada sintética e correspondente é bastante pequena: 1,1 mais ou menos 0,8 px (largura do olho = cinquenta e cinco px).

Figura 9. Distribua o gráfico das distâncias entre os centros de pupilas de imagens sintéticas e reais.



Como configurar os hiper-parâmetros? Dicas e truques.

Inicialização de G

Primeiro, inicializamos G apenas com a perda de auto-regularização para que possa começar a produzir uma versão desfavorável da entrada sintética. Normalmente, levou 500-2,000 passos de G (sem treinamento D).

Primeiro, inicializamos G apenas com a perda de auto-regularização para que possa começar a produzir uma versão desfavorável da entrada sintética. Normalmente, levava cincocentos a dois mil passos de G (sem treinamento D).

Diferentes passos de G e D para cada iteração de treino

Usamos diferentes números de etapas para o gerador e discriminador em cada iteração de treinamento. Para a estimativa da postura manual usando profundidade, usamos 2 etapas de G para cada passo D e, para o experimento de estimação do olho, acabamos usando 50 passos de G para cada etapa D. Achamos que o discriminador converge mais rapidamente em comparação com o gerador, em parte devido à norma do lote no discriminador. Então, nós corrigimos as etapas #D para 1 e começamos a variar as etapas #G a partir de um número pequeno, aumentando lentamente, dependendo dos valores de perda de discriminação.

Usamos diferentes números de etapas para o gerador e discriminador em cada iteração de treinamento. Para a estimativa da postura manual usando a profundidade, usamos 2 passos de G para cada passo D, e para o experimento de estimação do olhar nos olhos, acabamos usando cinquenta passos de G para cada etapa D. Achamos que o discriminador converge mais rapidamente em comparação com o gerador, em parte devido à norma do lote no discriminador. Então, nós corrigimos as etapas #D para 1 e começamos a variar as etapas #G a partir de um número pequeno, aumentando lentamente, dependendo dos valores de perda de discriminação.

Taxa de aprendizagem e critérios de parada

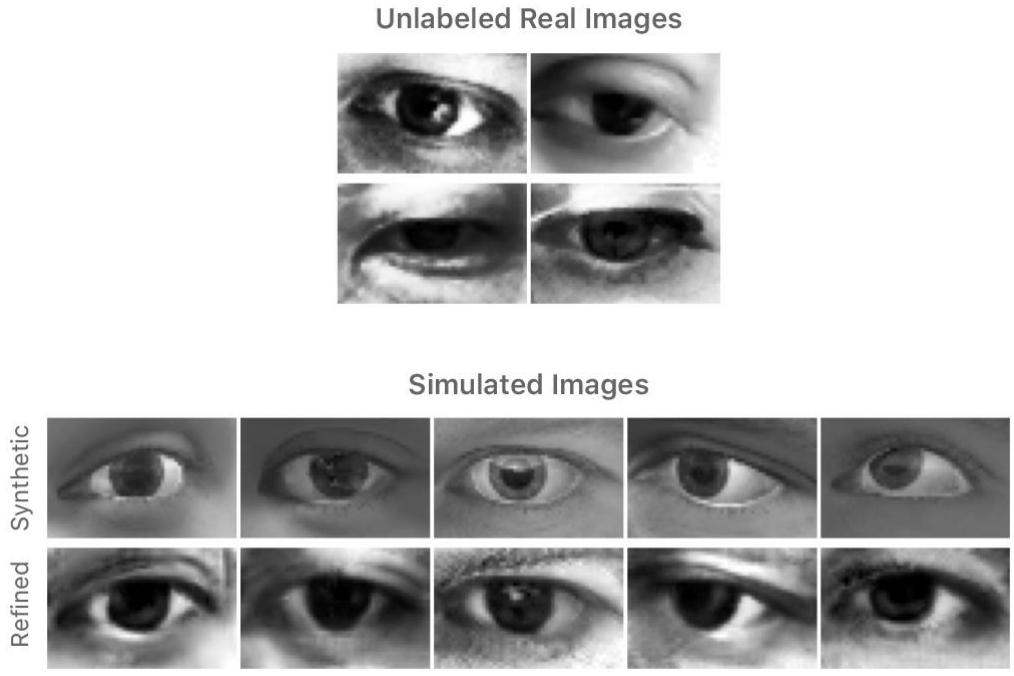
Achamos útil manter a taxa de aprendizagem muito pequena (~ 0,0001) e treinar por um longo período de tempo. Esta abordagem funcionou provavelmente porque ele mantém o gerador ou o discriminador de fazer mudanças repentinas que deixariam o outro para trás. Achamos difícil parar o treinamento, visualizando a perda de treinamento. Em vez disso, salvamos as imagens de treinamento à medida que o treinamento progride e parou de treinar quando as imagens refinadas pareciam visualmente semelhantes às imagens reais.

Resultados qualitativos

Para avaliar a qualidade visual das imagens refinadas, nós criamos um estudo de usuário simples onde os sujeitos foram convidados a classificar as imagens como sintéticas reais ou refinadas. Os assuntos achavam muito difícil distinguir as imagens reais e refinadas. Em nossa análise agregada, 10 indivíduos escolheram o rótulo correto 517 vezes em 1000 ensaios, o que significa que eles não foram capazes de distinguir de forma confiável imagens reais de sintéticas refinadas. Em contraste, ao testar imagens sintéticas originais versus imagens reais, mostramos 10 imagens sintáticas reais e 10 por assunto, e os sujeitos escolheram corretamente 162 vezes em 200 ensaios. Na Figura 10, mostramos alguns exemplos de imagens refinadas sintéticas e correspondentes.

Para avaliar a qualidade visual das imagens refinadas, nós criamos um estudo de usuário simples onde os sujeitos foram convidados a classificar as imagens como sintéticas reais ou refinadas. Os assuntos achavam muito difícil distinguir as imagens reais e refinadas. Em nossa análise agregada, dez indivíduos escolheram o rótulo correto quinhentos e dezessete vezes em 1000 ensaios, o que significa que eles não conseguiram distinguir de maneira confiável imagens reais de sintéticas refinadas. Em contraste, ao testar imagens sintéticas originais versus imagens reais, mostramos dez imagens reais e dez sintéticas por assunto, e os sujeitos escolheu corretamente cento e sessenta e duas vezes em duzentos tentativas. Na Figura dez, mostramos algumas imagens sintéticas e correspondentes refinadas.

Figura 10 dez . Exemplo de imagens de olhos refinados usando o método proposto.



Resultados Quantitativos

A Figura 11 mostra a melhoria usando dados refinados, em comparação com treinamento com dados sintéticos originais. Duas coisas a serem observadas a partir desta figura: (1) O treinamento com imagens refinadas é melhor do que o treinamento com imagens sintéticas originais, e (2) o uso de mais dados sintéticos melhora o desempenho. Na Figura 12, comparamos o erro de estimativa do olhar com outros métodos de última geração e mostramos que melhorar o realismo ajuda significativamente o modelo a generalizar dados de teste reais.

A figura 11 mostra a melhoria usando dados refinados, em comparação com treinamento com dados sintéticos originais. Duas coisas a serem observadas a partir desta figura: (1) O treinamento com imagens refinadas é melhor do que o treinamento com imagens sintéticas originais, e (2) o uso de mais dados sintéticos melhora o desempenho. Na Figura 12, comparamos o erro de estimativa do olhar com outros métodos de última geração e mostramos que melhorar o realismo ajuda significativamente o modelo a se generalizar em dados reais de teste.

Figura 11 onze . Comparação de treinamento usando imagens sintéticas e refinadas para estimativa de olhar. Avaliado em imagens de teste reais.

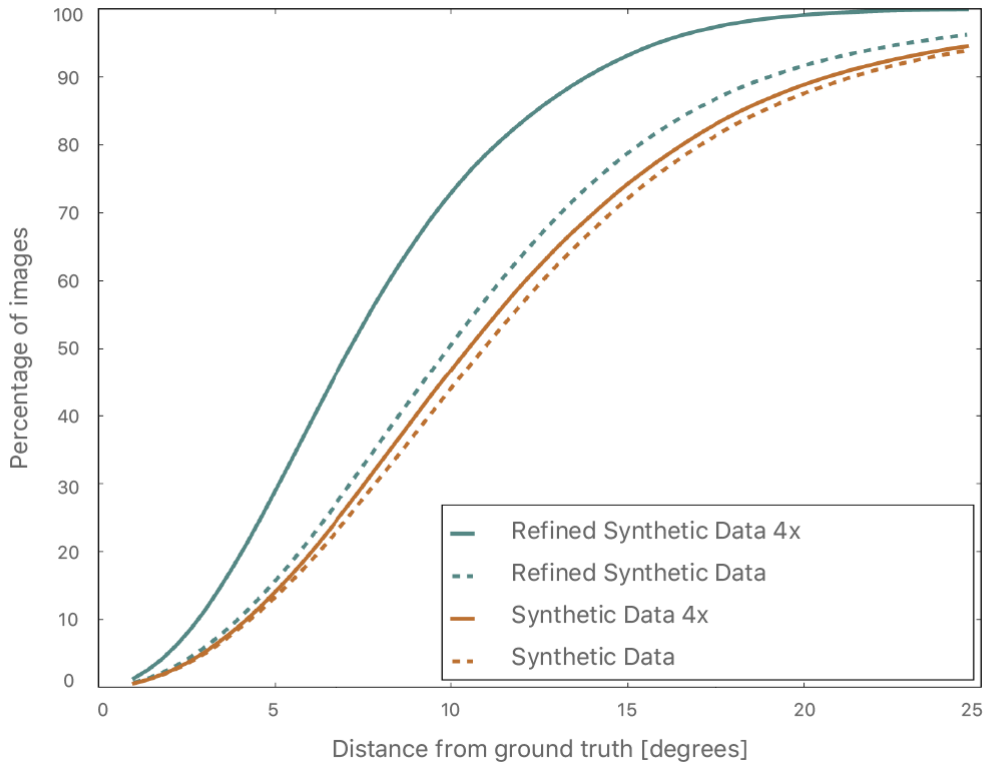


Figura 12 doze . Comparação de diferentes métodos para estimativa de olhar no conjunto de dados MPIIGaze. Os dois primeiros métodos são descritos nas referências [2] e [3] .

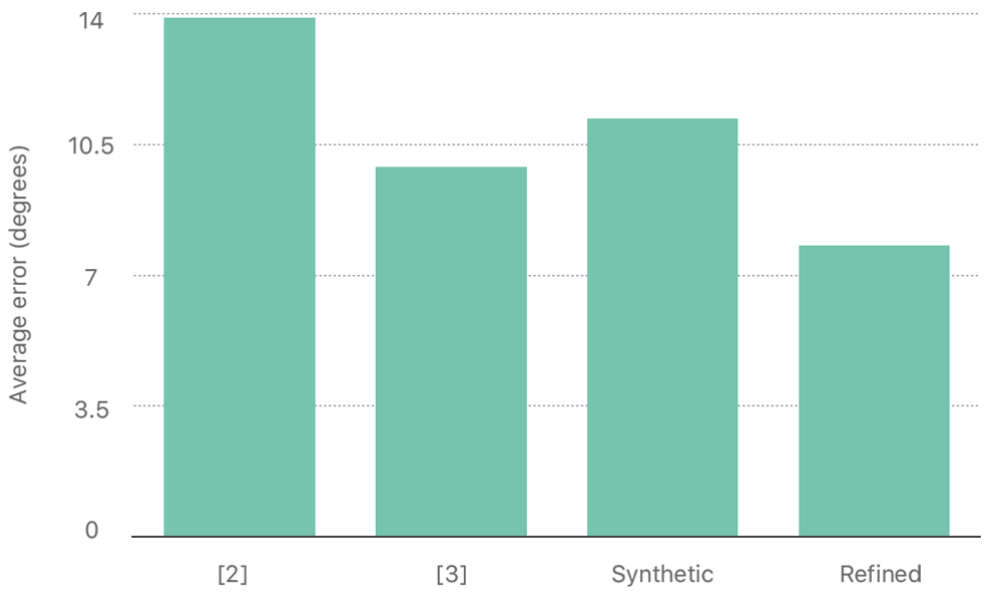


Figura 12 doze . Comparação de diferentes métodos para estimativa de olhar no conjunto de dados MPIIGaze. Os dois primeiros métodos são descritos nas referências [2] e [3] .

Trabalho relatado

Recentemente, tem tido muito interesse na adaptação do domínio usando treinamento adversarial. O trabalho de tradução de imagem para imagem [4] de Isola et al. Descreva um método que aprende a mudar uma imagem de um domínio para outro, mas precisa de correspondências em pixels. O papel de tradução sem imagem da imagem para imagem [5] discute relaxar o requisito de correspondência em pixels e segue nossa estratégia de usar o histórico do gerador para melhorar o discriminador. A rede de tradução de imagem para imagem não supervisionada [6] usa uma combinação de um auto-codificador GAN e variacional para aprender o mapeamento entre os domínios de origem e de destino. Costa et al. [7] usam idéias de nosso trabalho para aprender a gerar imagens do fundo do olho. Sela et al. [8] usam uma abordagem similar de auto-regularização para a reconstrução da geometria facial. Lee et al. [9] aprenda a sintetizar uma imagem a partir de patches locais chave usando um discriminador em patches. Para obter mais detalhes sobre o trabalho que descrevemos neste artigo, consulte nosso artigo CVPR "Aprendendo com imagens simuladas e não supervisionadas através de treinamento adverso" [10] .

Referências

[1] IJ Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville e Y. Bengio, **Generative Adversarial Nets** . *Proceedings Neural Information Processing Systems Conference* , 2014. dois mil e quatorze

[2] X. Zhang, Y. Sugano, M. Fritz e A. Bulling, Looke **-based Gaze Estimation in the Wild** . *Concurso de reconhecimento de padrões de visão de computador* , 2015. dois mil e quinze

[3] E. Wood, T. Baltrušaitis, L.-P. Morency, P. Robinson e A. Bulling, **aprendendo um Estimador Gaze baseado em aparência de um milhão de imagens sintetizadas** . *Proceedings ACM Symposium on Eye Tracking Research & Applications* , 2016. dois mil e dezesseis

[4] P. Isola, J.-Y. Zhu, T. Zhou e AA Efros, **Tradução Imagem-Imagem com Redes Adversas Condicionais** . ArXiv, 2016. dois mil e dezesseis

[5] J.-Y. Zhu, T. Park, P. Isola e AA Efros, **tradução sem imagem da Imagem-Imagem, usando Redes Adversas Consistentes ao Ciclo** . ArXiv, 2017. dois mil dezessete

[6] M.-Y. Liu, T. Breuel e J. Kautz, **Redes de Tradução Imagem-Imagem sem** Supervisão . ArXiv, 2017. dois mil dezessete

[7] P. Costa, A. Galdran, MI Meyer, MD Abràmoff, M. Niemeijer, AMMendonça e A. Campilho, **para a síntese de imagem retinal adversária** . ArXiv, 2017. dois mil dezessete

[8] M. Sela, E. Richardson e R. Kimmel, **Reconstrução de Geometria Facial Não Restrita Usando a Tradução de Imagem para Imagem** . ArXiv, 2017. dois mil dezessete

[9] D. Lee, S.Yun, S. Choi, H. Yoo, M.-H. Yang e S. Oh, **geração de imagem holística não supervisionada a partir de patches locais chave** . ArXiv, 2017. dois mil dezessete

[10] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, R. Webb, **aprendendo de imagens simuladas e não supervisionadas através de treinamento adverso** . CVPR, 2017.