Part II
Generative Models for Unsupervised Learning

Paolo Favaro

Computer Vision Group
Computer Science Department - University of Bern

Unsupervised Learning

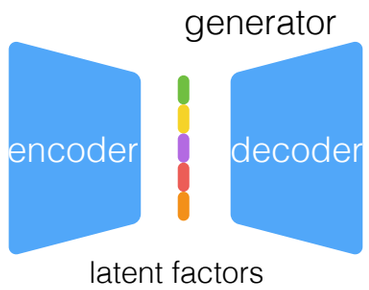
- We could use self-supervised learning, but it captures only **part** of the structure in the data
  - We might be less likely to do well on a new task
  - Then combine several SSL tasks via **multi-task learning**
  - Not straightforward
- Alternative: capture **all** the structure of the data

Training a Generative Model

- To capture all the structure of the data we consider an **encoder** and a **decoder/generator**

- The encoder identifies the latent factors in a certain data sample

- The decoder/generator is a **generative model** that renders data from the representation

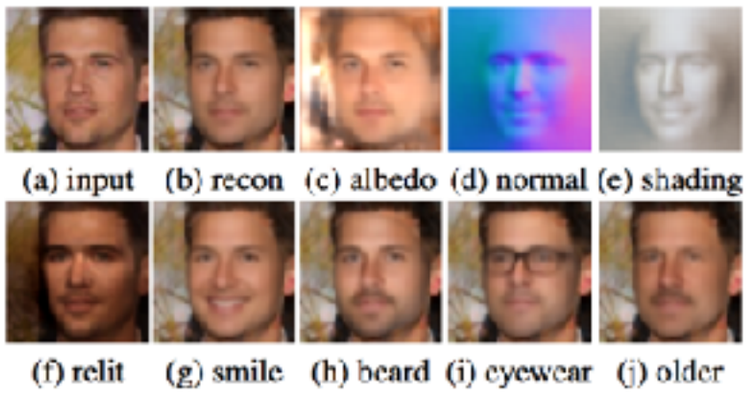


Reconstruction Constraint

- Can be achieved via an **autoencoding** constraint
- Can be achieved through a **generative adversarial network** (i.e., by matching the distribution of the generated samples to that of the given samples)
- Recent methods also introduced models of image formation/differential renderers

Example of Model-Based Reconstruction

- Learn a representation that factorizes components of an image formation model (normals, albedo, shading, and matte layer)

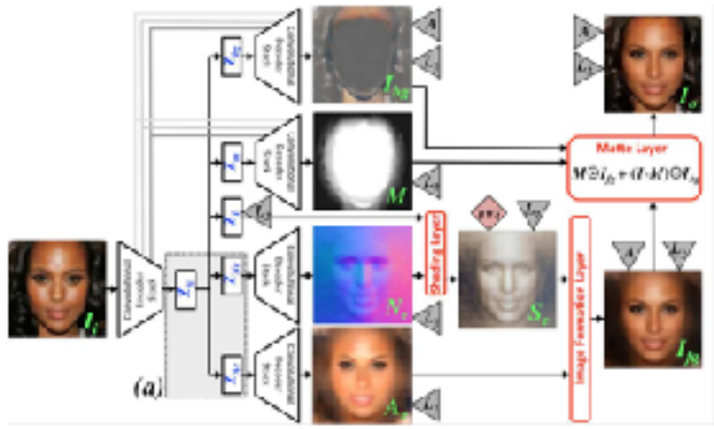


Z. Shu, E. Yumer, S. Hadap, K. Sunkavalli, E. Shechtman, and D. Samaras. Neural face editing with intrinsic image disentangling. CVPR 2017

Uses 3D model to help ill-posedness of model

Face Editing

- The model recovers all terms in the model and the loss function imposes that the model should reconstruct the input image

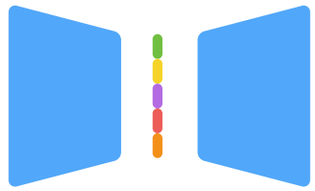


Z. Shu, E. Yumer, S. Hadap, K. Sunkavalli, E. Shechtman, and D. Samaras. Neural face editing with intrinsic image disentangling. CVPR 2017

Applications

- The generative framework leads to two main applications
  - **Editing**: Since we can encode and generate the data, then we can combine the features of different samples to generate a new sample
  - **Transfer learning**: The learned representation groups factors of variations, which should transfer well to new tasks

Training a Generative Model

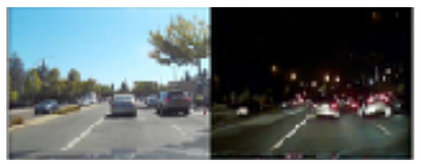


- The main building block is an autoencoding model
- What properties do we want to impose to the latent representation?
- To impose these properties we need to rely on data and possibly available (weak) labeling
- **Two scenarios**: One with weak annotation (with/no human annotation) and one fully unsupervised

Disentangling Factors with Weak Annotation

Weak Annotation
Can Come for Free with the Setup

- Data is captured so that only one factor of variation changes
- Examples
  - Images from a stationary webcam, where the viewpoint is the same and what changes is the time of capture (day/night/different seasons/weather)



  - Images from a synchronized stereo or multiview rig: the scene, time, illumination etc is all the same except the viewpoint

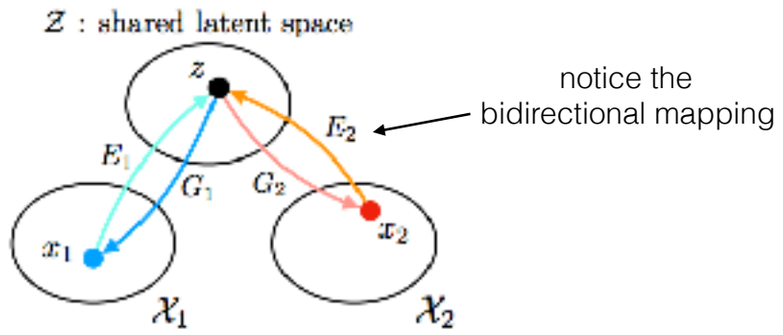
UINT: Unsupervised Image-to-Image Translation Networks

- Suppose that we have samples from domain $\mathcal{X}_1$ and from domain $\mathcal{X}_2$ (this is a form of weak labeling)

M. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. NIPS 2017

UINT: Unsupervised Image-to-Image Translation Networks

- Suppose that we have samples from domain $\mathcal{X}_1$ and from domain $\mathcal{X}_2$ (this is a form of weak labeling)

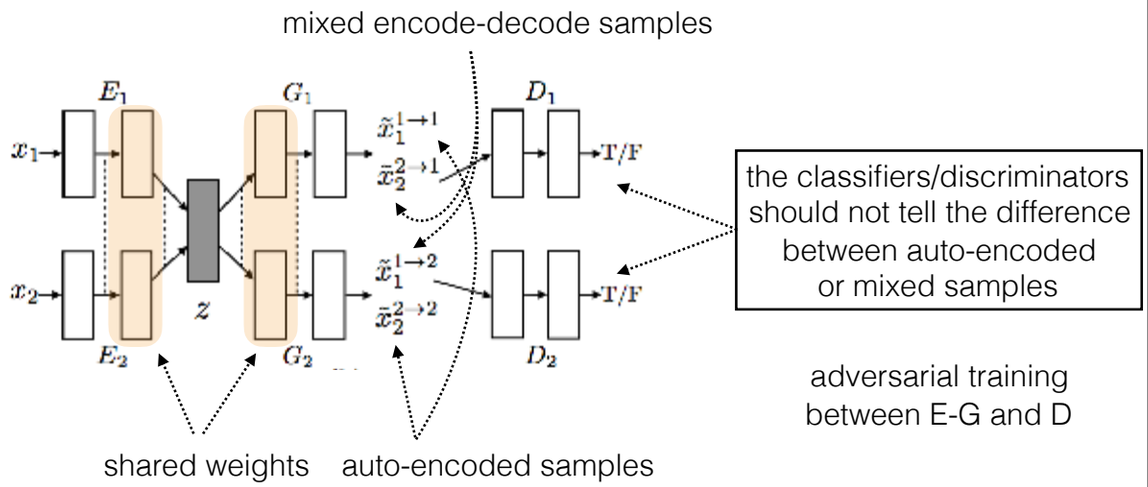


we would like to map them to a shared latent space

M. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. NIPS 2017

UINT: Unsupervised Image-to-Image Translation Networks

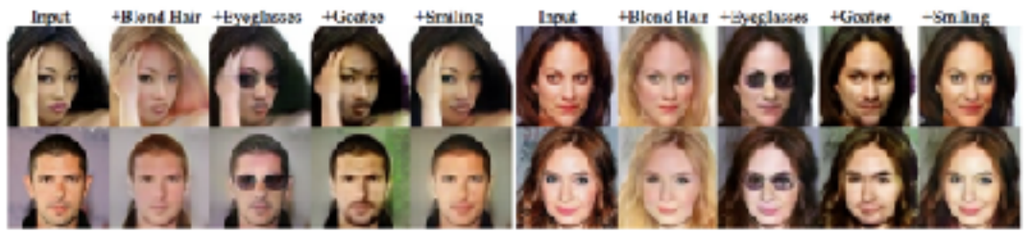
- Force latent space sharing by mixing the encoding and decoding of samples from the two domains



M. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. NIPS 2017

Editing with UINT

Dog breeds translation results



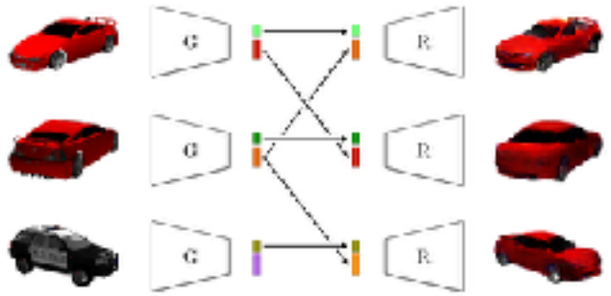
Attribute-based face translation results.

M. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. NIPS 2017

Pros: - able to translate attributes/breeds between the same category
Cons: - complicated network and not easy to train

Disentangling Varying and Common Factors

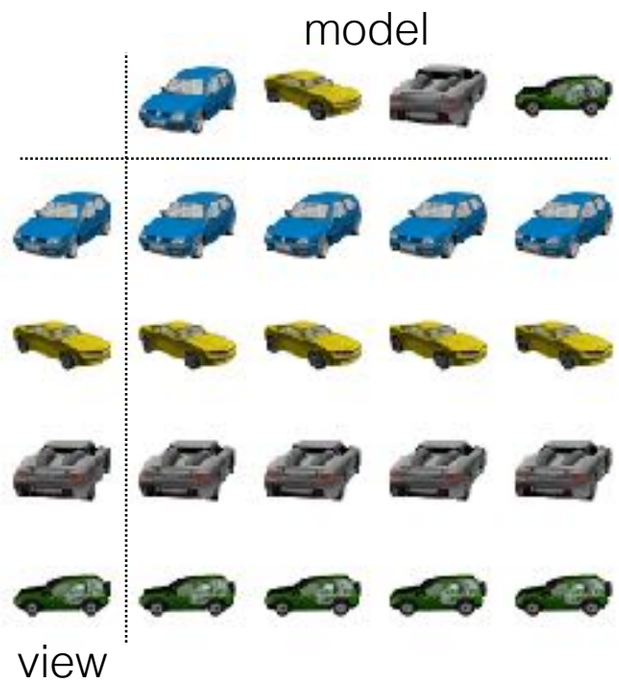
- Consider images of cars from a stereo camera
- We do not know what the viewpoint of the car in each stereo image is (varying factor), but we know that the car is the same (common factor)



disentangle pose and car ID

Challenges in Disentangling Independent Factors of Variation
A. Szabó, Q. Hu, T. Portenier, M. Zwicker, P. Favaro, ECCV 2018

Shortcut Problem

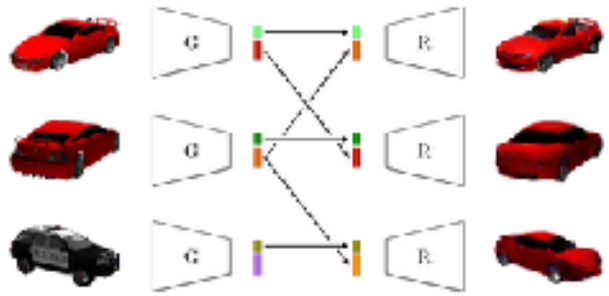


- the common attribute is not transferred
- the encoder puts all the information in the varying factor
- simple solution: reduce the dimension of the varying factor (then it can't encode all variability)
- not desirable: we don't know the "correct" dimensionality a priori

Disentangling Varying and Common Factors

- **Ambiguity**: Not possible to guarantee that factors use the same reference system for each attribute

- E.g., the pose feature of a corvette may have nothing to do with the pose feature of a porsche



disentangle pose and car ID

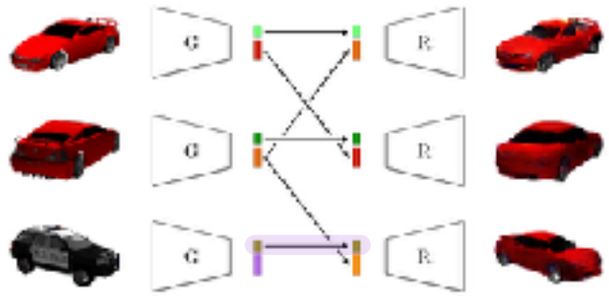
- However, in practice these models learn representations with the same reference system

Challenges in Disentangling Independent Factors of Variation
A. Szabó, Q. Hu, T. Portenier, M. Zwicker, P. Favaro, ECCV 2018

Disentangling Varying and Common Factors

- **Ambiguity**: Not possible to guarantee that factors use the same reference system for each attribute

- E.g., the pose feature of a corvette may have nothing to do with the pose feature of a porsche

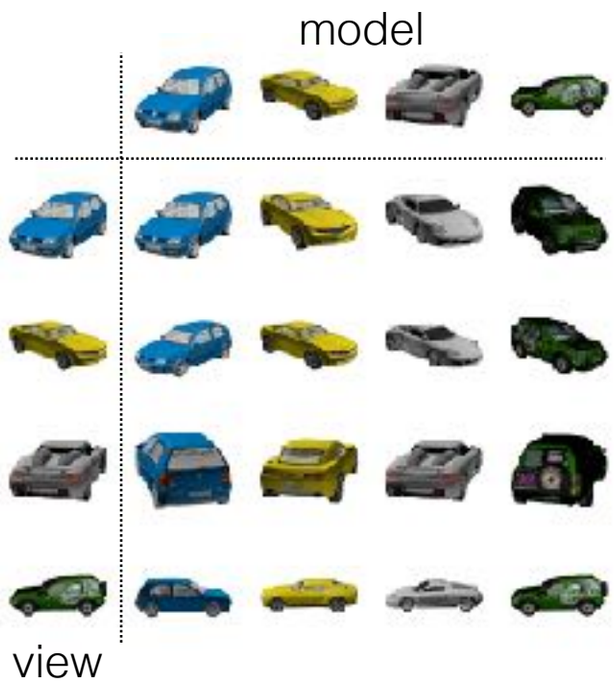


disentangle pose and car ID

- However, in practice these models learn representations with the same reference system

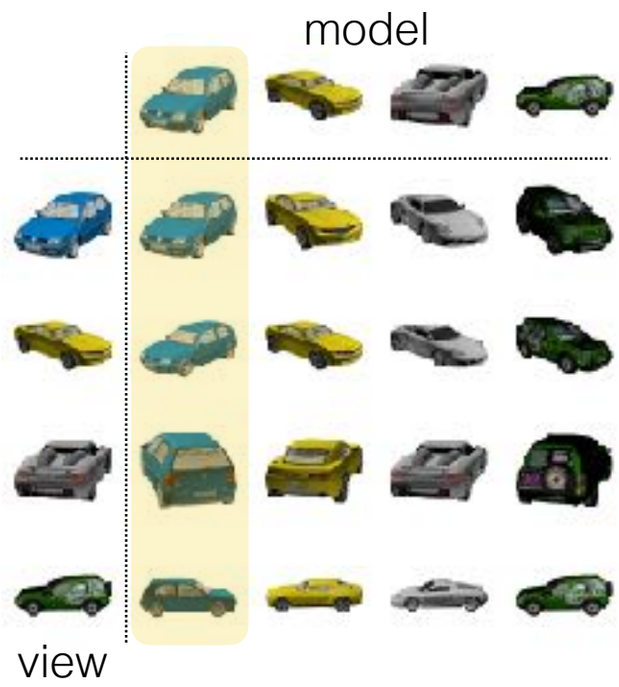
Challenges in Disentangling Independent Factors of Variation
A. Szabó, Q. Hu, T. Portenier, M. Zwicker, P. Favaro, ECCV 2018

Reference Ambiguity



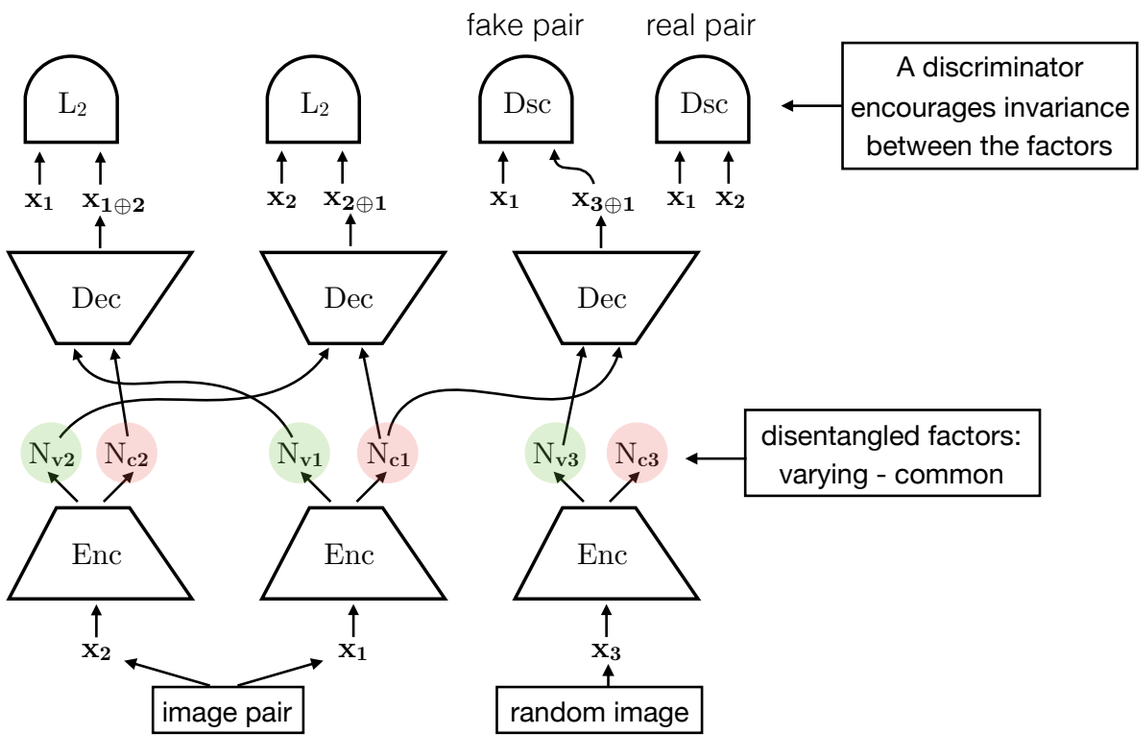
- the viewpoint interpretation depends on the car type
- the reference is clockwise for the blue car and counterclockwise for the others
- this ambiguity cannot be avoided

Reference Ambiguity

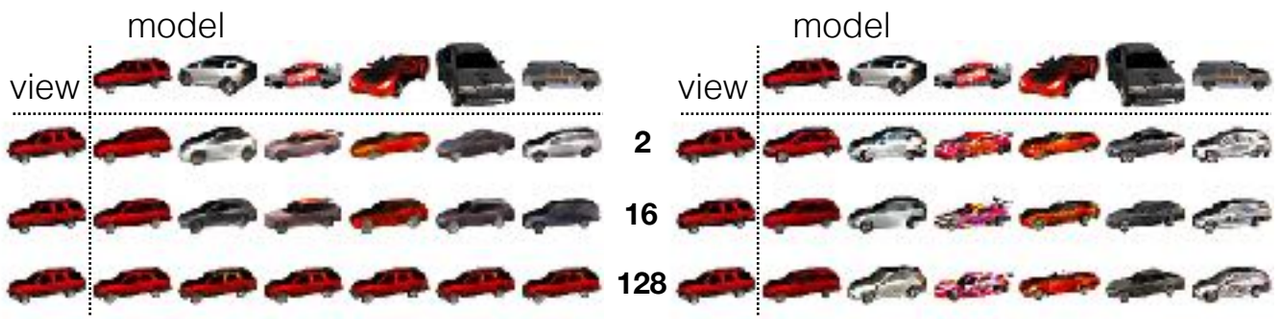


- the viewpoint interpretation depends on the car type
- the reference is clockwise for the blue car and counterclockwise for the others
- this ambiguity cannot be avoided

Architecture



Shortcut Problem
Transfer Experiments



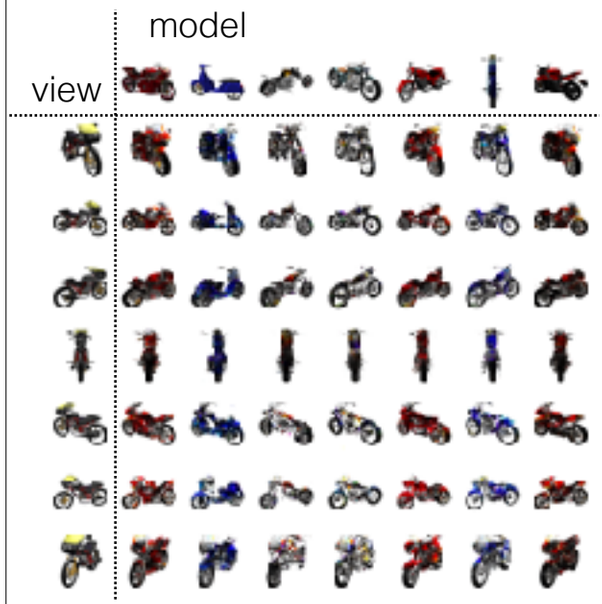
variable factor dimension

(swap) Autoencoder

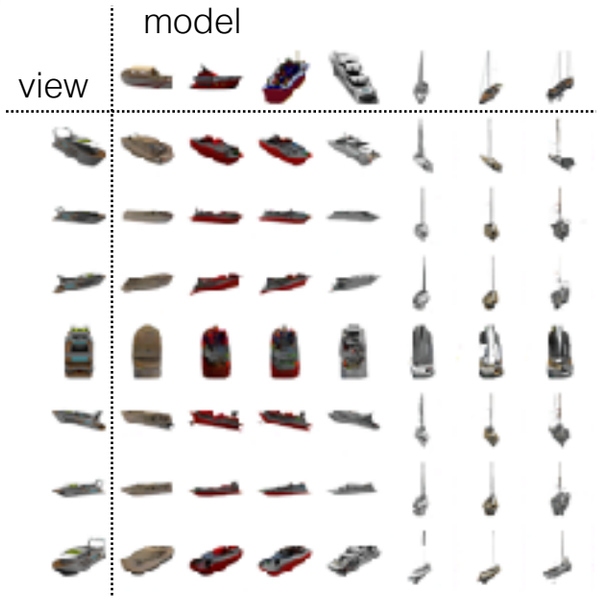
(swap) Autoencoder + GAN

Reference Ambiguity

Transfer Experiments



bikes: no ambiguity



vessels: ambiguity is present

Disentangling Factors without Annotation

Unsupervised Disentangling of Factors of Variation

- We would like to learn a feature representation that separates feature attributes

without using attribute labels (or other knowledge)

Unsupervised Disentangling of Factors of Variation

- We would like to learn a feature representation that separates feature attributes

without using attribute labels (or other knowledge)

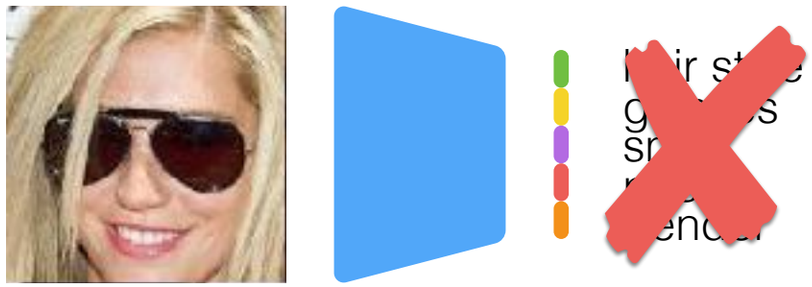
Unsupervised Disentangling of Factors of Variation

- We would like to learn a feature representation that separates feature attributes

without using attribute labels (or other knowledge)

Unsupervised Disentangling of Factors of Variation

- We would like to learn a feature representation that separates feature attributes



without using attribute labels (or other knowledge)

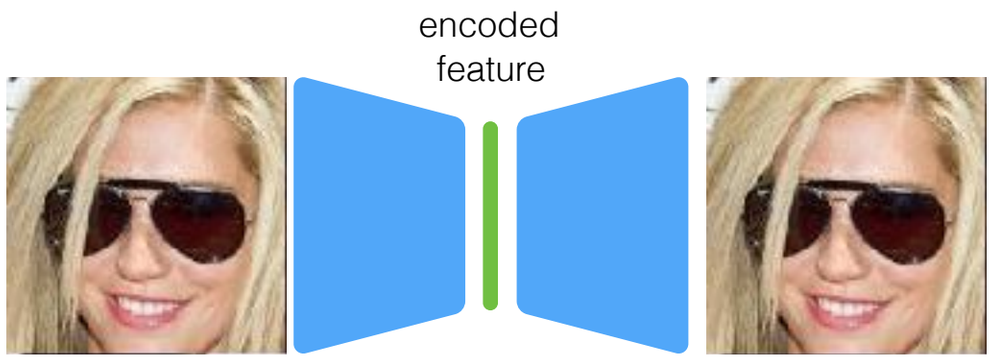
Unsupervised Disentanglement

- **Desiderata**
  - Avoid any labels, just use dataset samples
  - Learn an encoder to several (unknown) attributes
  - Learn a decoder to real images
  - Copy-paste from images to transfer attributes

Disentangling Factors of Variation by Mixing Them,
Q. Hu, A. Szabo´, T. Portenier, P. Favaro, M. Zwicker. CVPR 2018

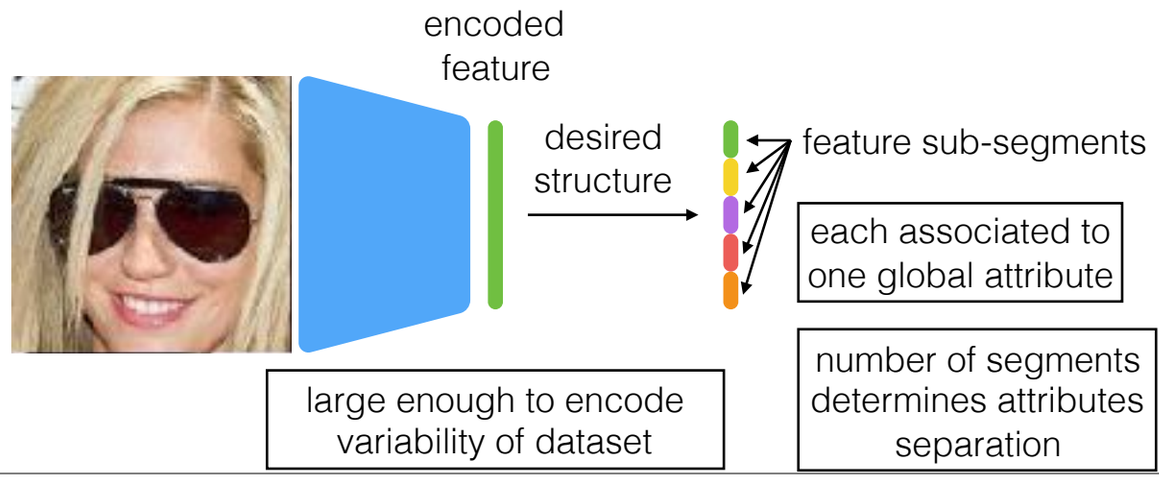
Feature Representation

- First, we choose an autoencoding architecture to preserve information of the input



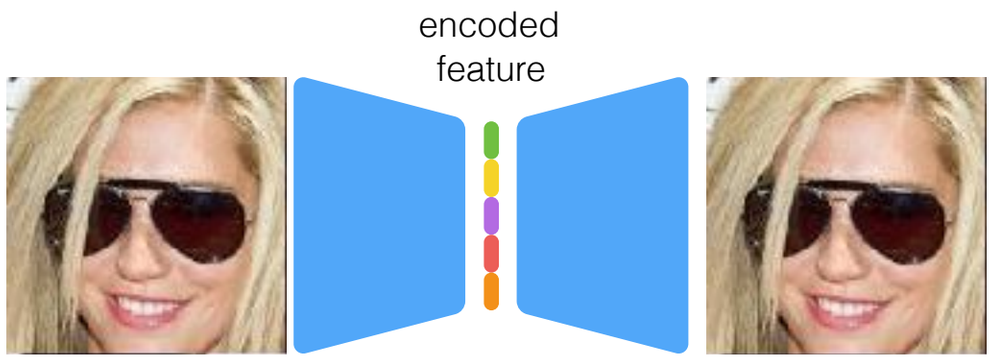
Feature Representation

- We want to split the encoded feature into several sub-segments each assigned to a global attribute



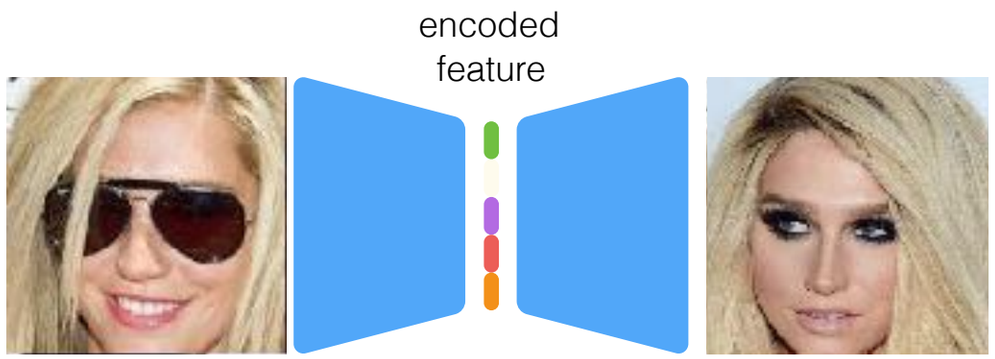
Feature Representation

- Feature changes should result in realistic images



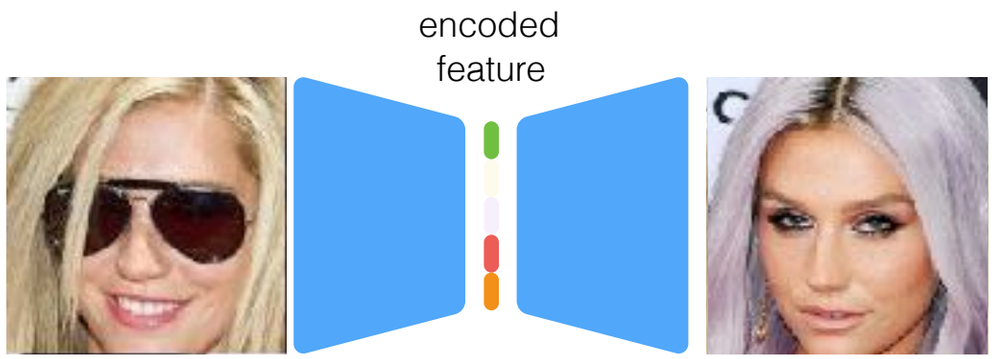
Feature Representation

- Feature changes should result in realistic images



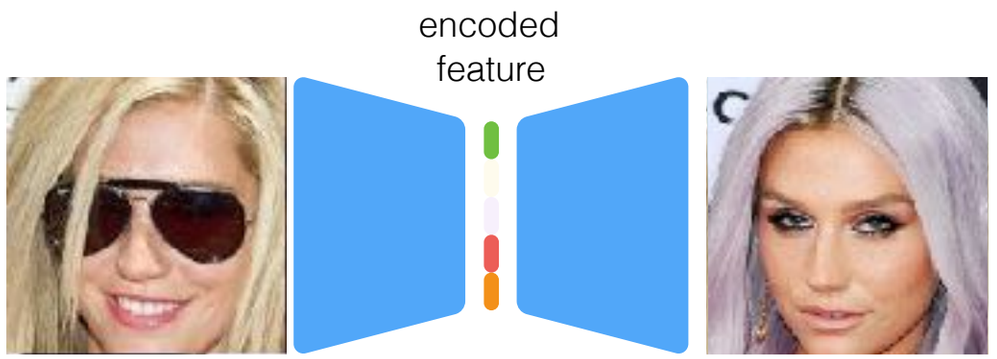
Feature Representation

- Feature changes should result in realistic images



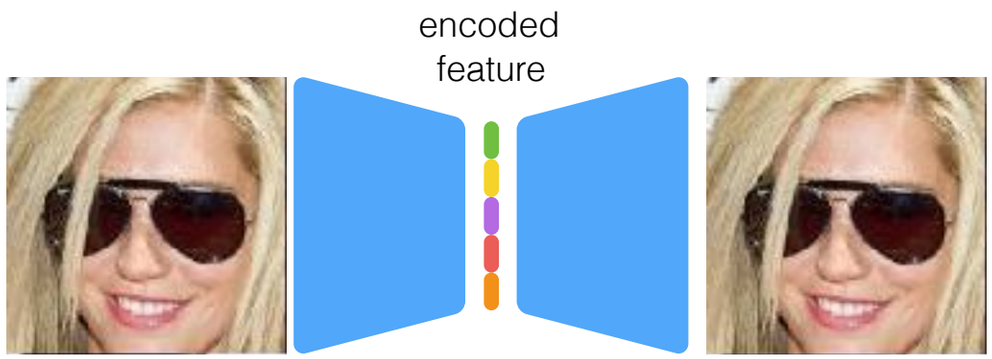
Feature Representation

- Feature changes should result in realistic images



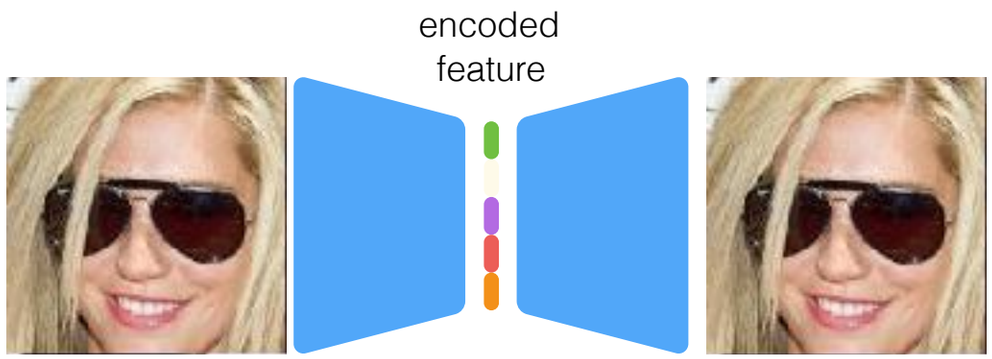
Feature Representation

- How to ensure that each change results in a different image?



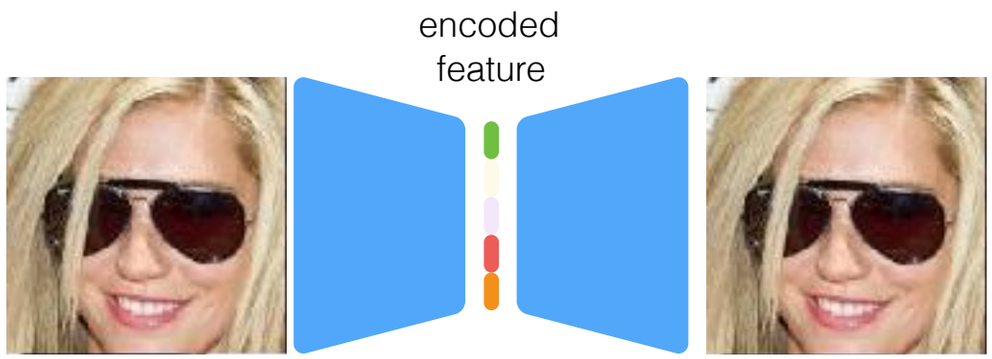
Feature Representation

- How to ensure that each change results in a different image?



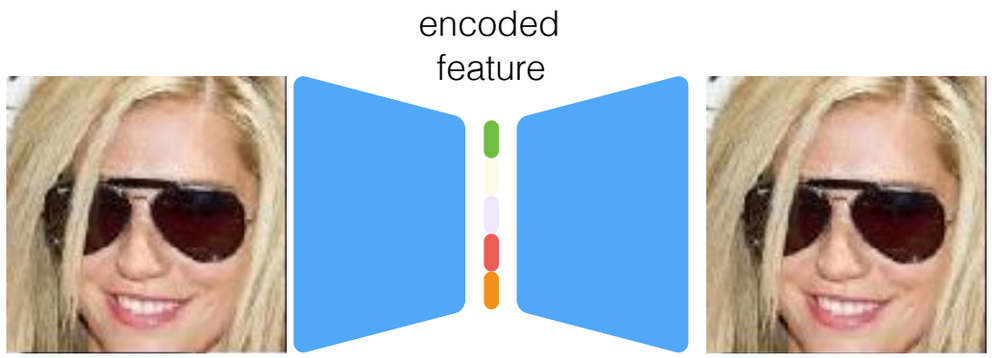
Feature Representation

- How to ensure that each change results in a different image?



Feature Representation

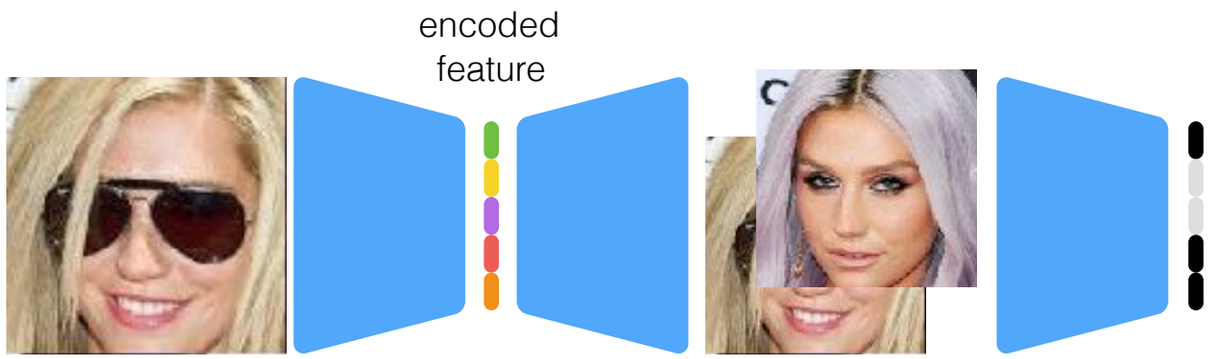
- How to ensure that each change results in a different image?



some segments in the feature could be ignored by the decoder

Feature Representation

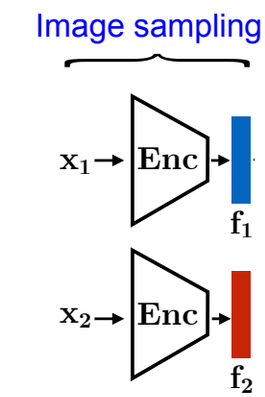
- How to ensure that each change results in a different image?



use a classifier to determine which segment was changed

Unsupervised Disentanglement

Network Architecture

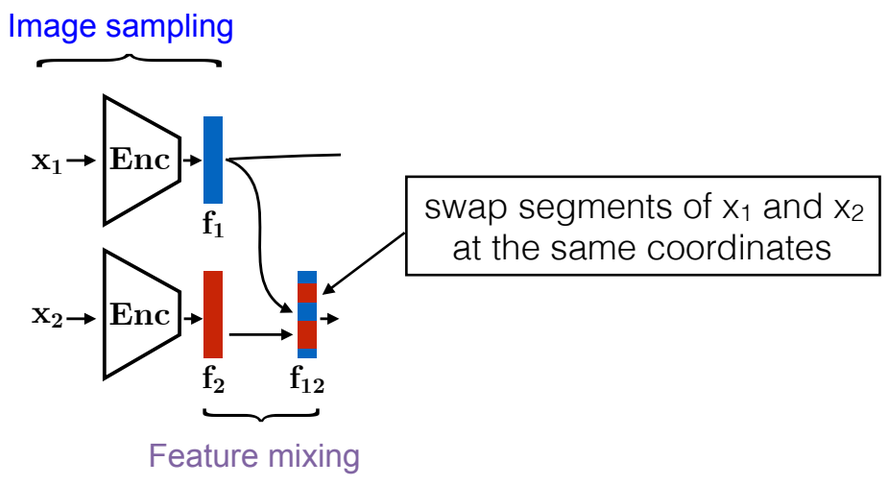


unrelated samples

Disentangling Factors of Variation by Mixing Them,
Q. Hu, A. Szabo´, T. Portenier, P. Favaro, M. Zwicker. CVPR 2018

Unsupervised Disentanglement

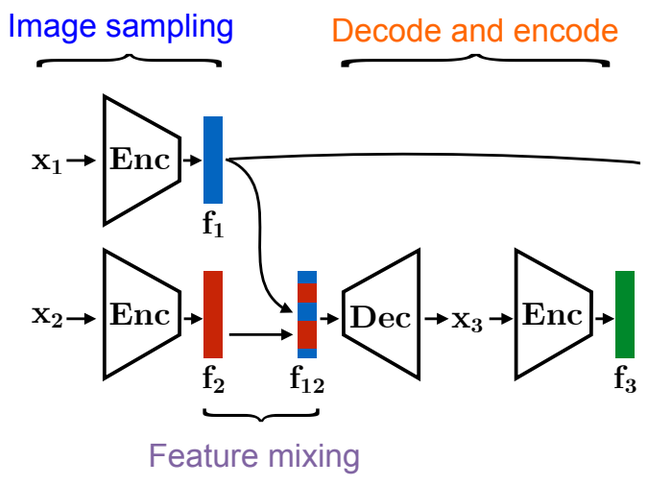
Network Architecture



Disentangling Factors of Variation by Mixing Them,
Q. Hu, A. Szabo´, T. Portenier, P. Favaro, M. Zwicker. CVPR 2018

Unsupervised Disentanglement

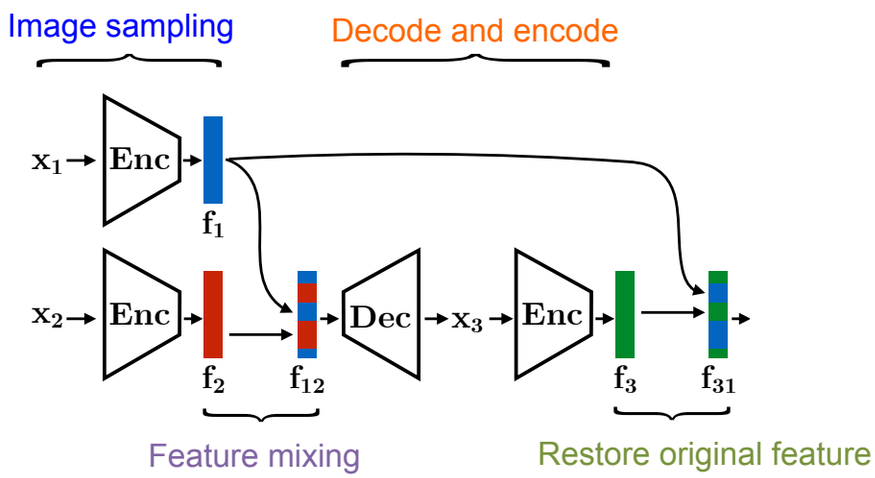
Network Architecture



Disentangling Factors of Variation by Mixing Them,
Q. Hu, A. Szabo´, T. Portenier, P. Favaro, M. Zwicker. CVPR 2018

Unsupervised Disentanglement

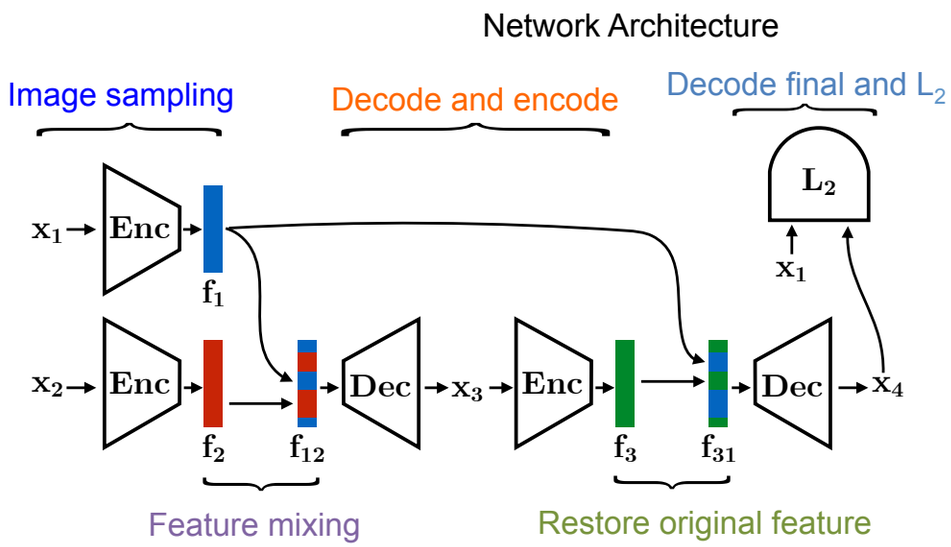
Network Architecture



Disentangling Factors of Variation by Mixing Them,
Q. Hu, A. Szabo´, T. Portenier, P. Favaro, M. Zwicker. CVPR 2018

Impose cycle consistency on the generated features

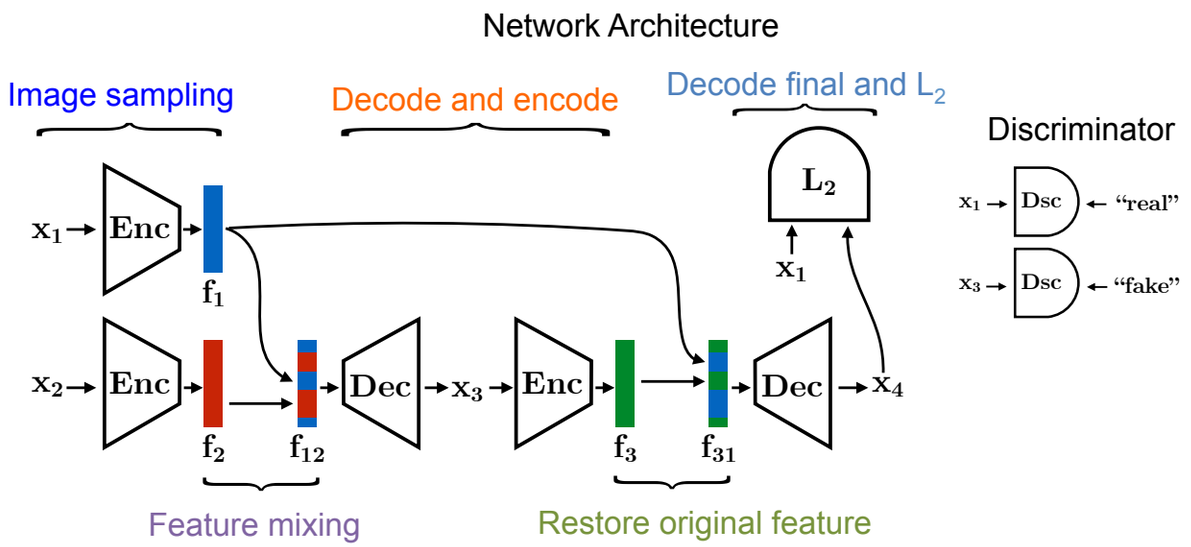
Unsupervised Disentanglement



Disentangling Factors of Variation by Mixing Them,
Q. Hu, A. Szabo´, T. Portenier, P. Favaro, M. Zwicker. CVPR 2018

Decode final and L2: Image cycle consistency

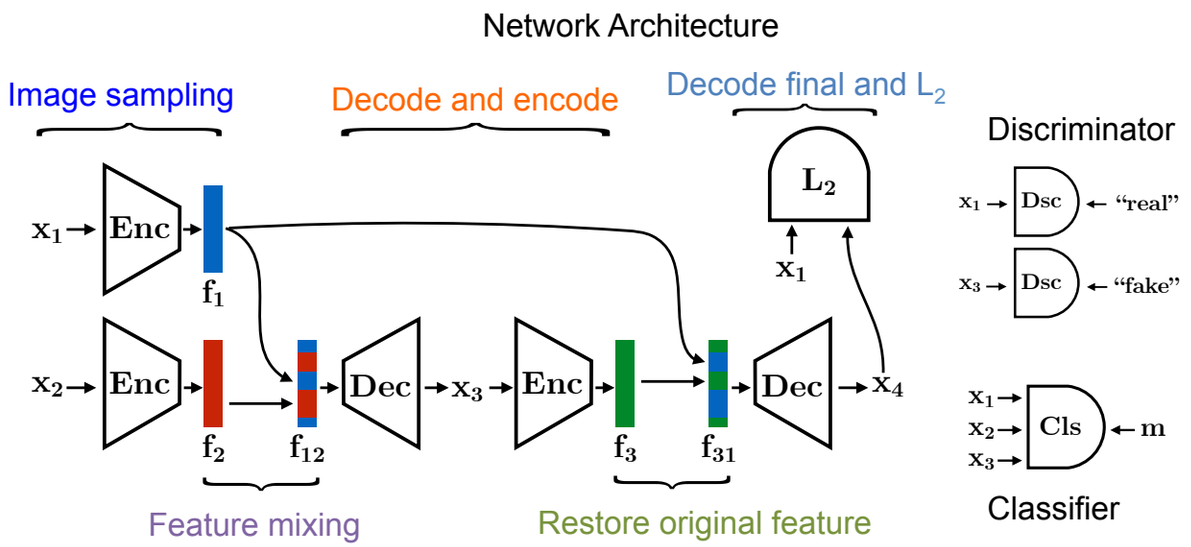
Unsupervised Disentanglement



Disentangling Factors of Variation by Mixing Them,
Q. Hu, A. Szabo´, T. Portenier, P. Favaro, M. Zwicker. CVPR 2018

Discriminator: Make sure that x3 looks real

Unsupervised Disentanglement



Disentangling Factors of Variation by Mixing Them,
Q. Hu, A. Szabo´, T. Portenier, P. Favaro, M. Zwicker. CVPR 2018

Classifier: make sure that all attributes are used

Experiments

Post-Identification of the Attributes

Pose+arm

Attribute transfer on SPRITES

Pros: - unsupervised method to disentangle the factors of variation
- transfer attributes between images
- effective on clustering attributes among images
- able to handle the big chunk size and avoid shortcut problem

Cons: - cannot handle the unaligned images

Experiments

Post-Identification of the Attributes

Skin+hair

Attribute transfer on SPRITES

Pros: - unsupervised method to disentangle the factors of variation
- transfer attributes between images
- effective on clustering attributes among images
- able to handle the big chunk size and avoid shortcut problem

Cons: - cannot handle the unaligned images

Experiments

Post-Identification of the Attributes

Vest

Attribute transfer on SPRITES

Pros: - unsupervised method to disentangle the factors of variation
- transfer attributes between images
- effective on clustering attributes among images
- able to handle the big chunk size and avoid shortcut problem

Cons: - cannot handle the unaligned images

Experiments

Post-Identification of the Attributes

Leg

Attribute transfer on SPRITES

Pros: - unsupervised method to disentangle the factors of variation
- transfer attributes between images
- effective on clustering attributes among images
- able to handle the big chunk size and avoid shortcut problem

Cons: - cannot handle the unaligned images

Experiments

Post-Identification of the Attributes

CelebA

Unsupervised Learning of 3D from an Image Collection

Given a dataset of real images with:
1) No 2 views of the same object instance
2) No annotation; e.g., landmarks, 3D templates, viewpoints

Goal
Learn to map 1 image with 1 object to its **3D**, **texture** and **viewpoint**

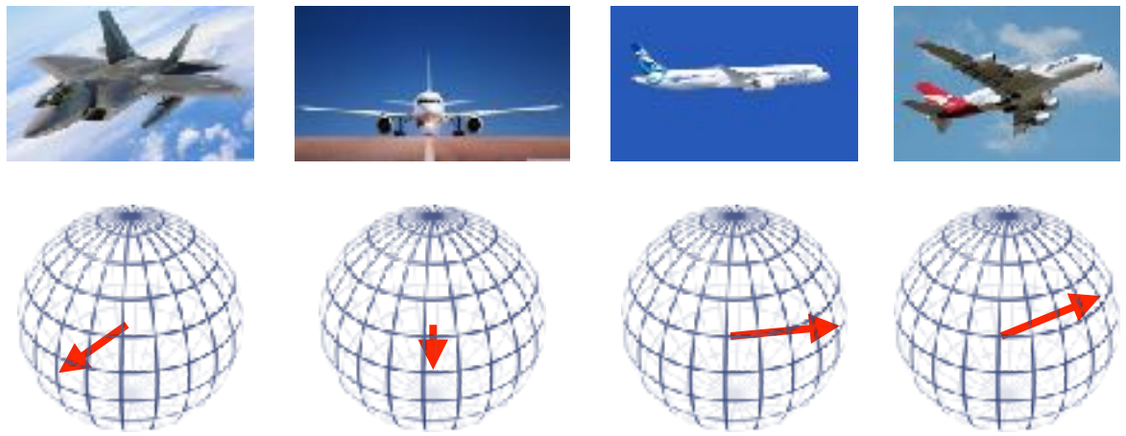
Unsupervised Learning of 3D from an Image Collection

Given a dataset of real images with:
1) No 2 views of the same object instance
2) No annotation; e.g., landmarks, 3D templates, viewpoints

Goal
Learn to map 1 image with 1 object to its **3D**, **texture** and **viewpoint**

A first step
Learn to map 1 image with 1 object to its **viewpoint**

Unsupervised Viewpoint Estimation



Szabó, Vedaldi and Favaro, *Building the View Graph of a Category by Exploiting Image Realism*, 3dRR, ICCV 2015

Unsupervised Viewpoint Estimation

compare images globally

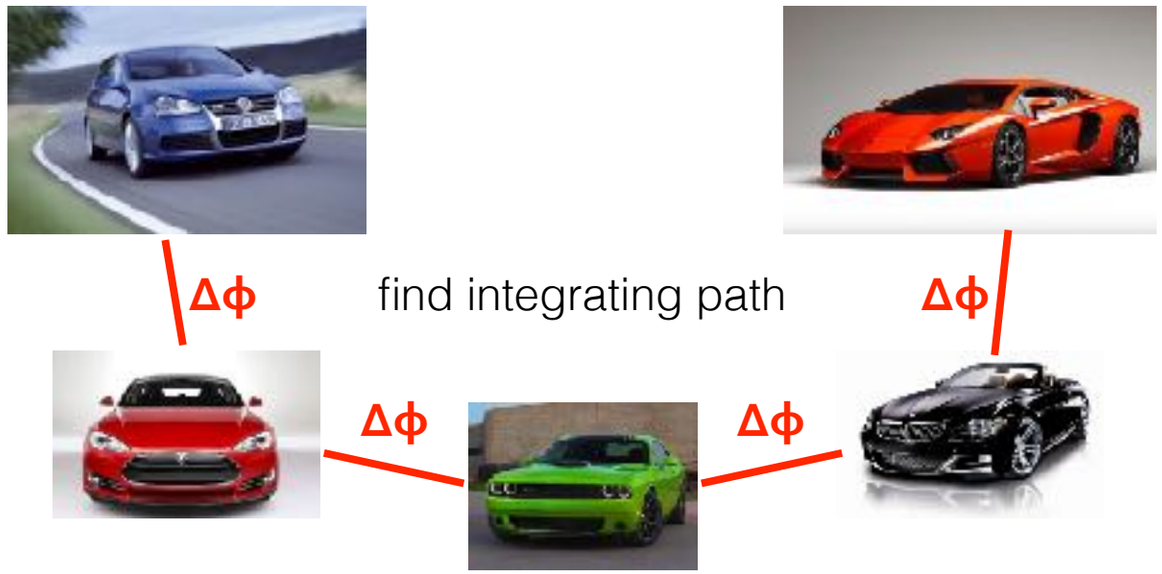
Estimate Relative Viewpoints

Δϕ

estimate small viewpoint changes

Δϕ

Estimate Relative Viewpoints



Results

Results

Key Ideas: 3D Hypothesis, Viewpoint Intervention and Realism

original images along the diagonal

Unsupervised Learning of 3D from an Image Collection

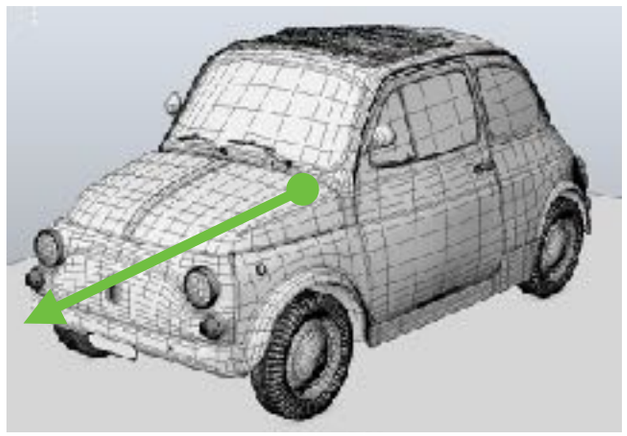
Given a dataset of real images with:
1) No 2 views of the same object instance
2) No annotation; e.g., landmarks, 3D templates, viewpoints

Goal
Learn to map 1 image with 1 object to its **3D**, **texture** and **viewpoint**

A. Szabó and P. Favaro, “Unsupervised 3D Shape Learning from Image Collections in the Wild”, arXiv 2018

UL of 3D from an Image Collection

Map 1 image with 1 object to its 3D, texture and viewpoint



Challenges

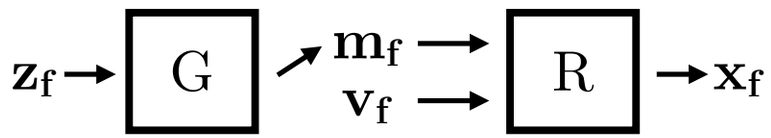
- Correspondence across images is not known
- Images cannot be matched directly (no stereo pairs nor video sequence)
- No specific prior knowledge of the objects is available (no 3D template, no category/attributes)
- Would like to deal with any mix of images (eg different categories)

A 3D Generative Model

- We propose to find correspondence through a 3D generative model
  - The generative model starts from noise and generates a representation of the object and the scene: 3D, texture and background
  - The representation is fed to a differentiable renderer with a random pose to generate an image

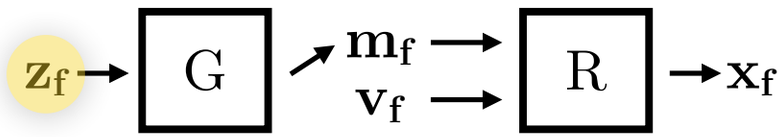
A 3D Generative Model

- If the generator G generates the correct 3D, texture and background, then if we render it from a random viewpoint, it should look realistic



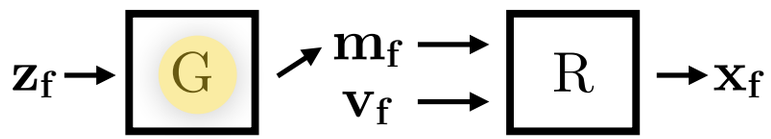
A 3D Generative Model

- If the generator G generates the correct 3D, texture and background, then if we render it from a random viewpoint, it should look realistic



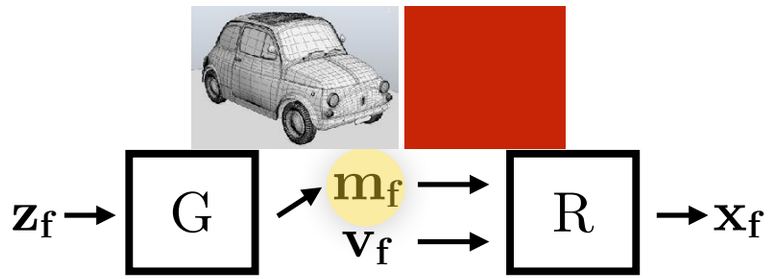
A 3D Generative Model

- If the generator G generates the correct 3D, texture and background, then if we render it from a random viewpoint, it should look realistic



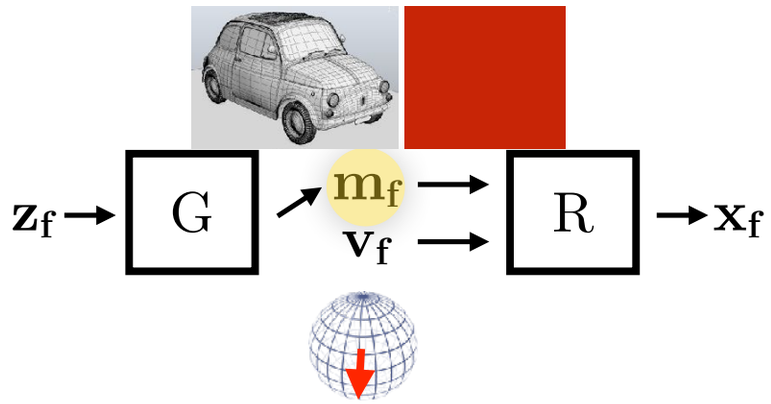
A 3D Generative Model

- If the generator G generates the correct 3D, texture and background, then if we render it from a random viewpoint, it should look realistic



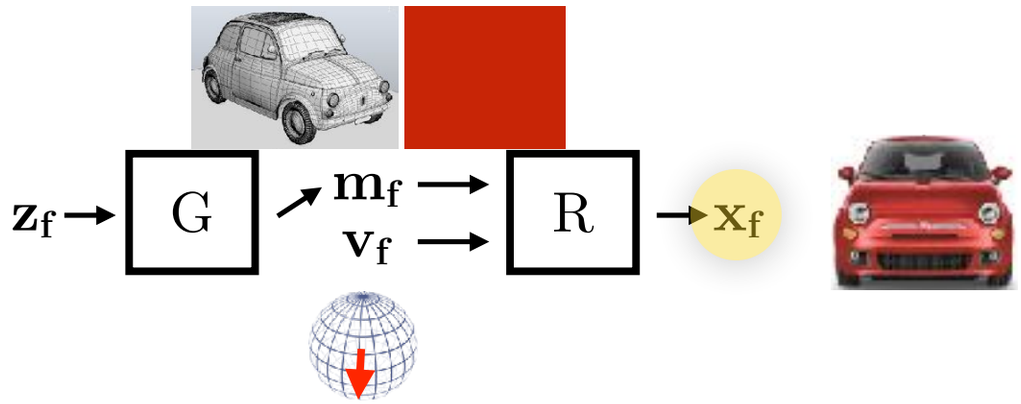
A 3D Generative Model

- If the generator G generates the correct 3D, texture and background, then if we render it from a random viewpoint, it should look realistic



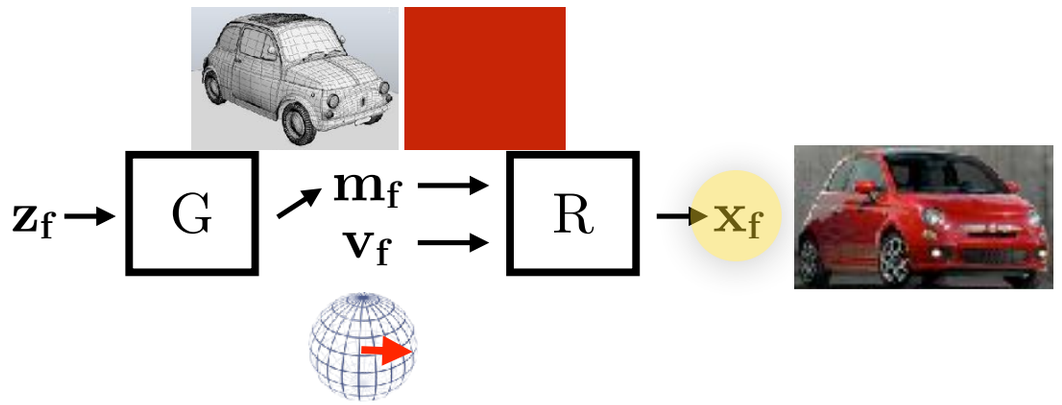
A 3D Generative Model

- If the generator G generates the correct 3D, texture and background, then if we render it from a random viewpoint, it should look realistic



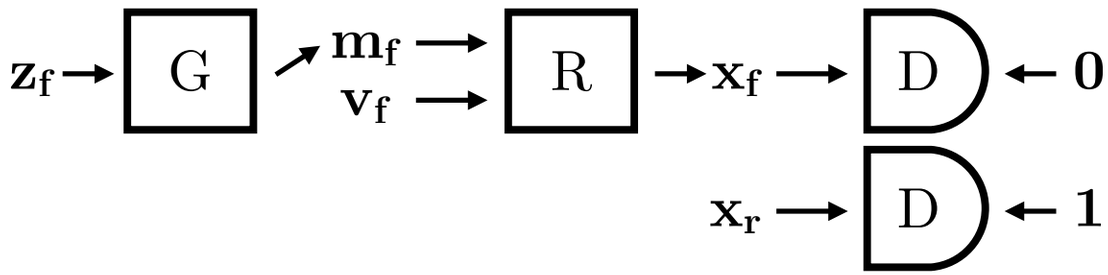
A 3D Generative Model

- If the generator G generates the correct 3D, texture and background, then if we render it from a random viewpoint, it should look realistic



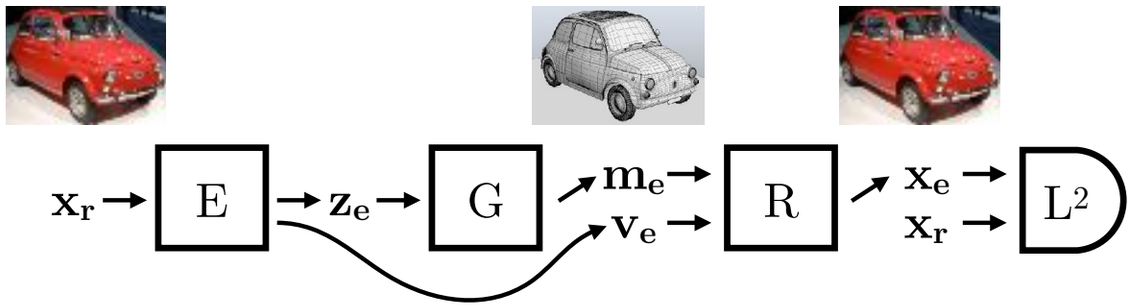
Enforcing Realism

- The 3D generative model is validated through intervention: We modify the pose and demand that the generated image be realistic
- Realism can be achieved through GAN training



Mapping Images to 3D and Pose

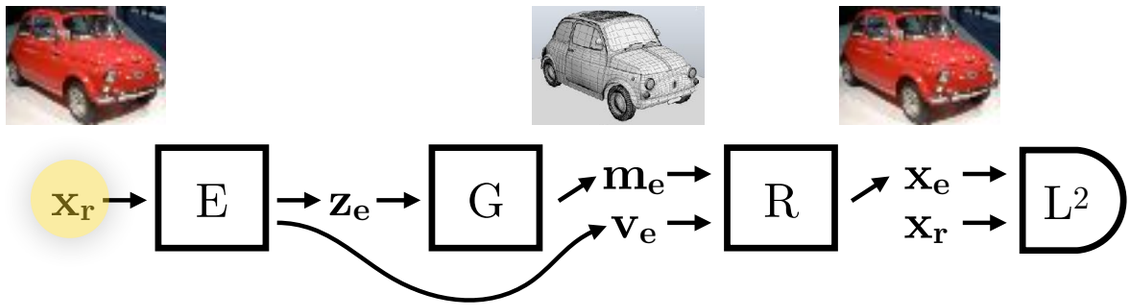
- We can also combine an encoder with the previous generator to map images to themselves (autoencoding)



- This makes the encoder learn how to map images to their 3D, texture, pose and background

Mapping Images to 3D and Pose

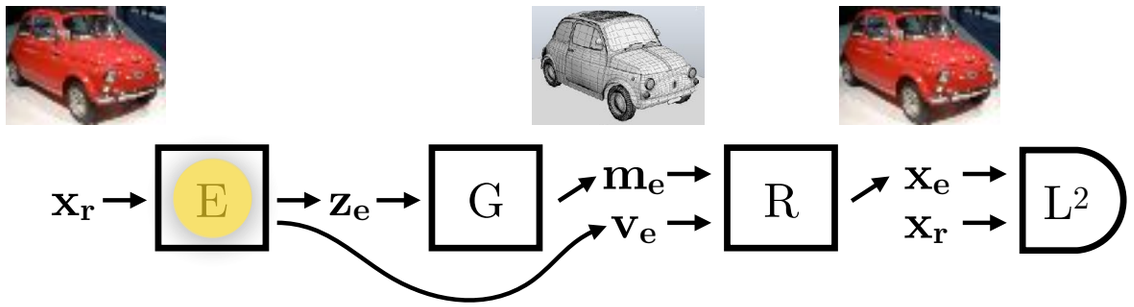
- We can also combine an encoder with the previous generator to map images to themselves (autoencoding)



- This makes the encoder learn how to map images to their 3D, texture, pose and background

Mapping Images to 3D and Pose

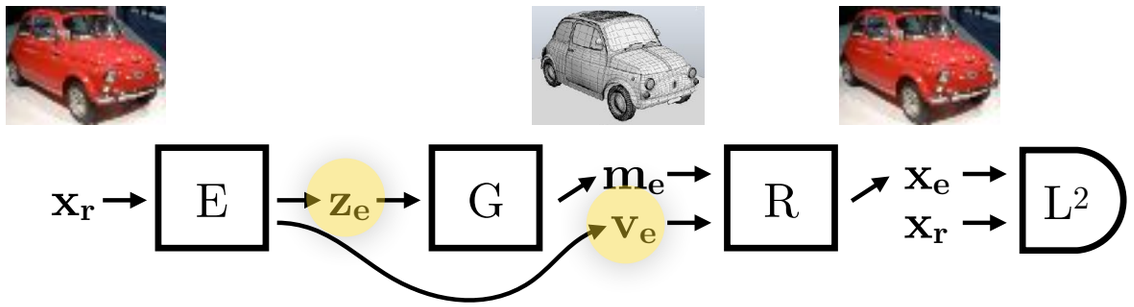
- We can also combine an encoder with the previous generator to map images to themselves (autoencoding)



- This makes the encoder learn how to map images to their 3D, texture, pose and background

Mapping Images to 3D and Pose

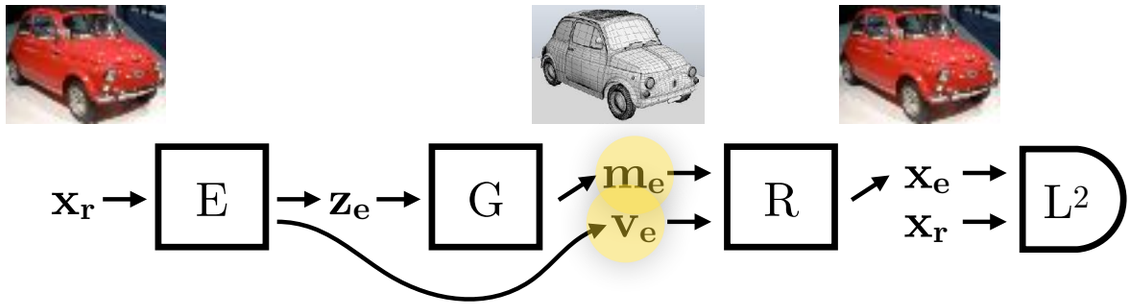
- We can also combine an encoder with the previous generator to map images to themselves (autoencoding)



- This makes the encoder learn how to map images to their 3D, texture, pose and background

Mapping Images to 3D and Pose

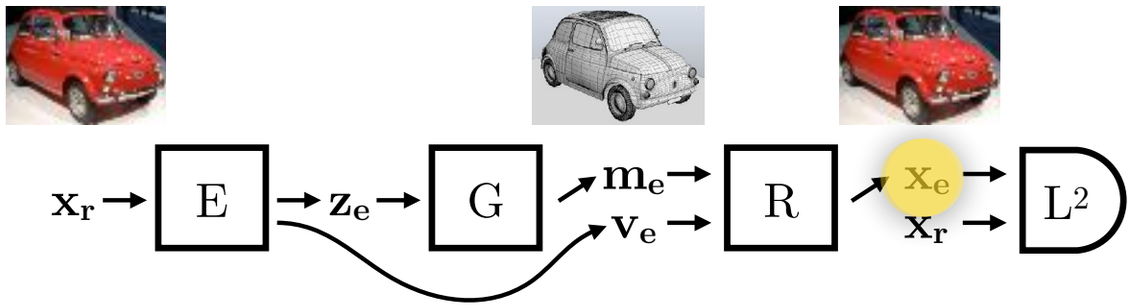
- We can also combine an encoder with the previous generator to map images to themselves (autoencoding)



- This makes the encoder learn how to map images to their 3D, texture, pose and background

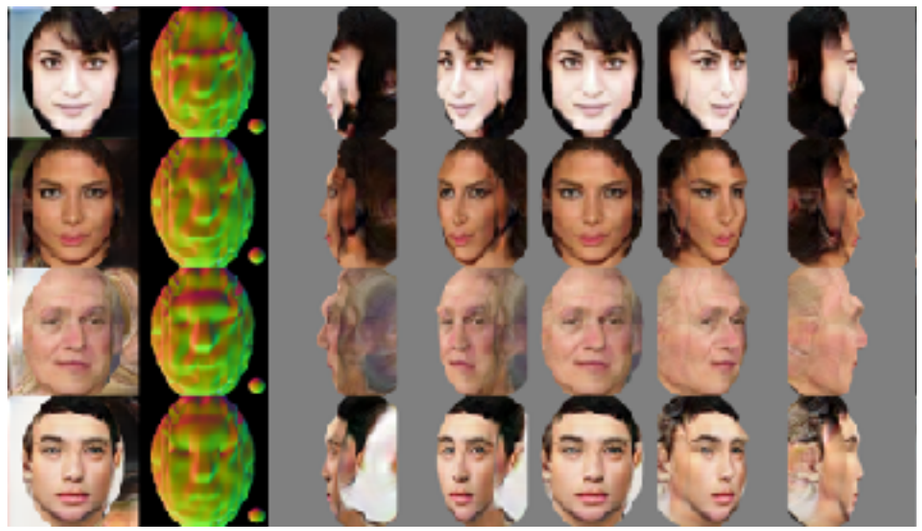
Mapping Images to 3D and Pose

- We can also combine an encoder with the previous generator to map images to themselves (autoencoding)



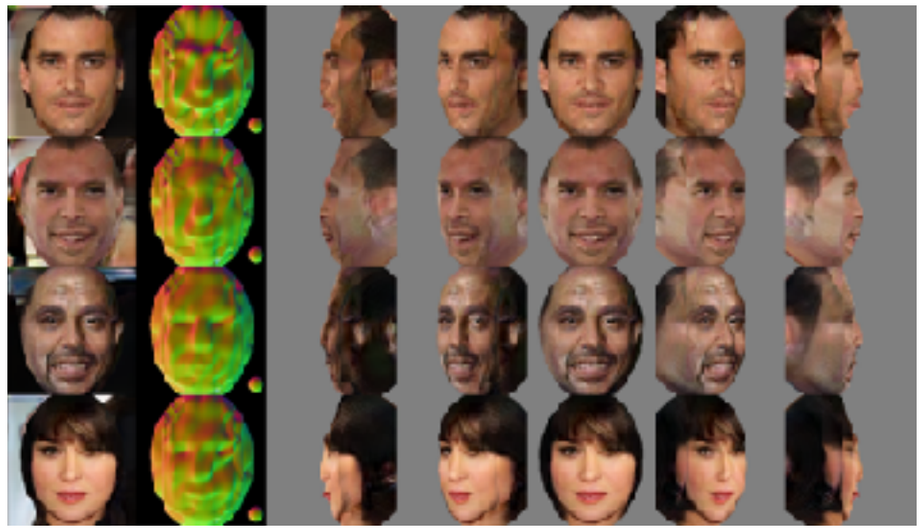
- This makes the encoder learn how to map images to their 3D, texture, pose and background

Generative Model on CelebA



output image | 3D | views without background

Generative Model on CelebA

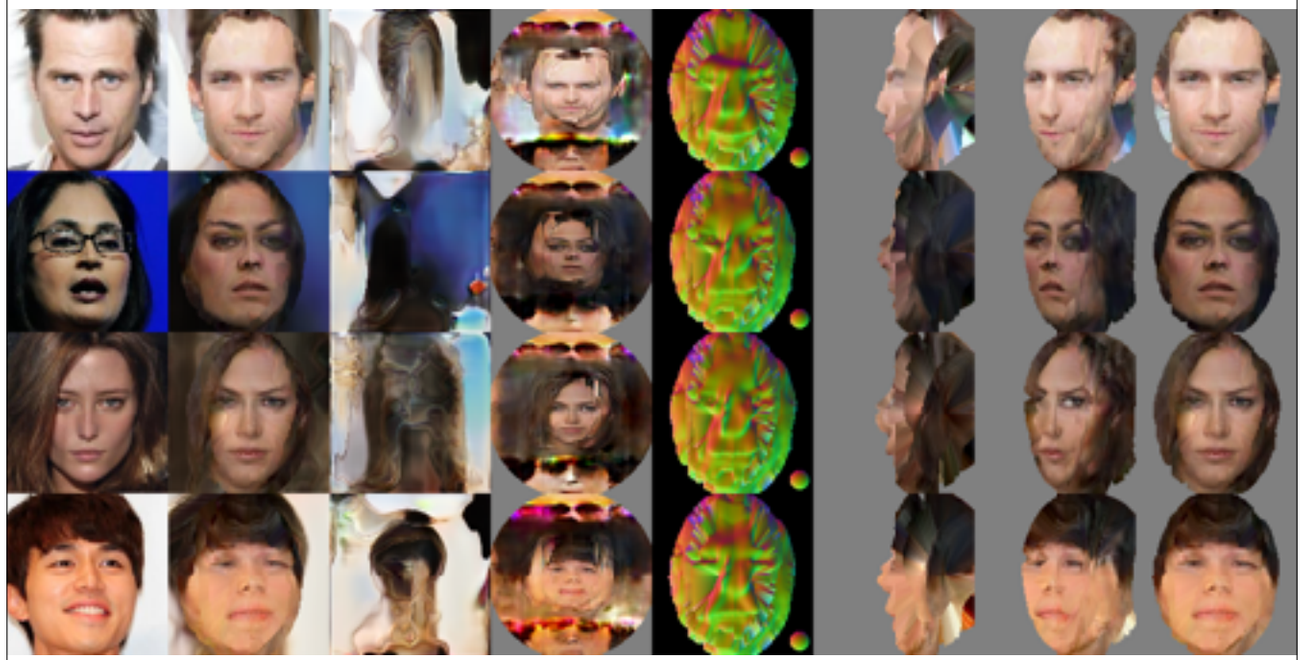


output image | 3D | views without background

Autoencoder on CelebA

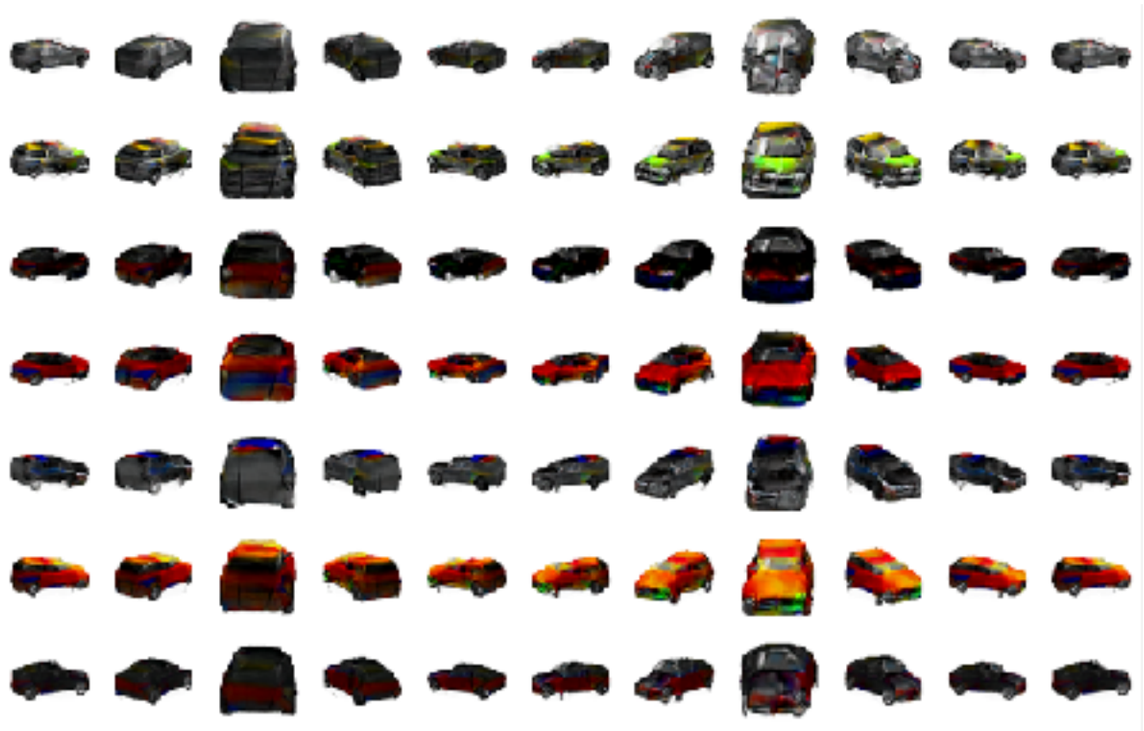
input reconstruction | background texture | 3D | views without background

Autoencoder on CelebA

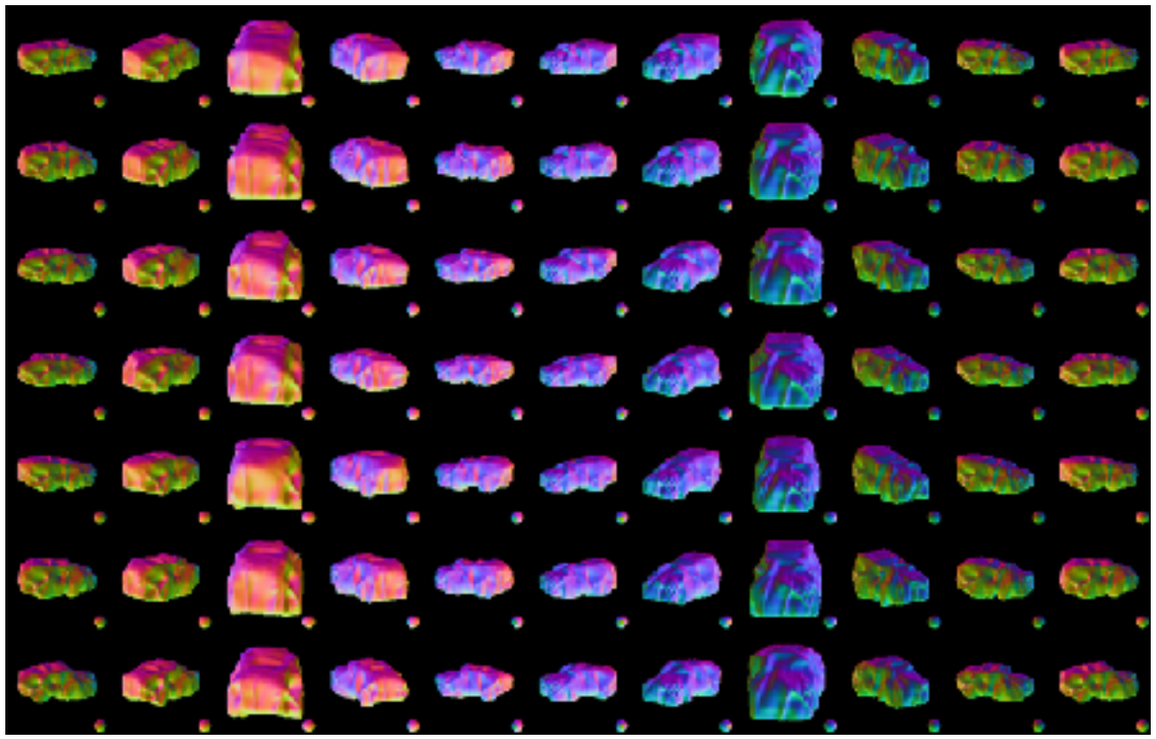


input reconstruction | background texture | 3D | views without background

GAN on ShapeNet - Car

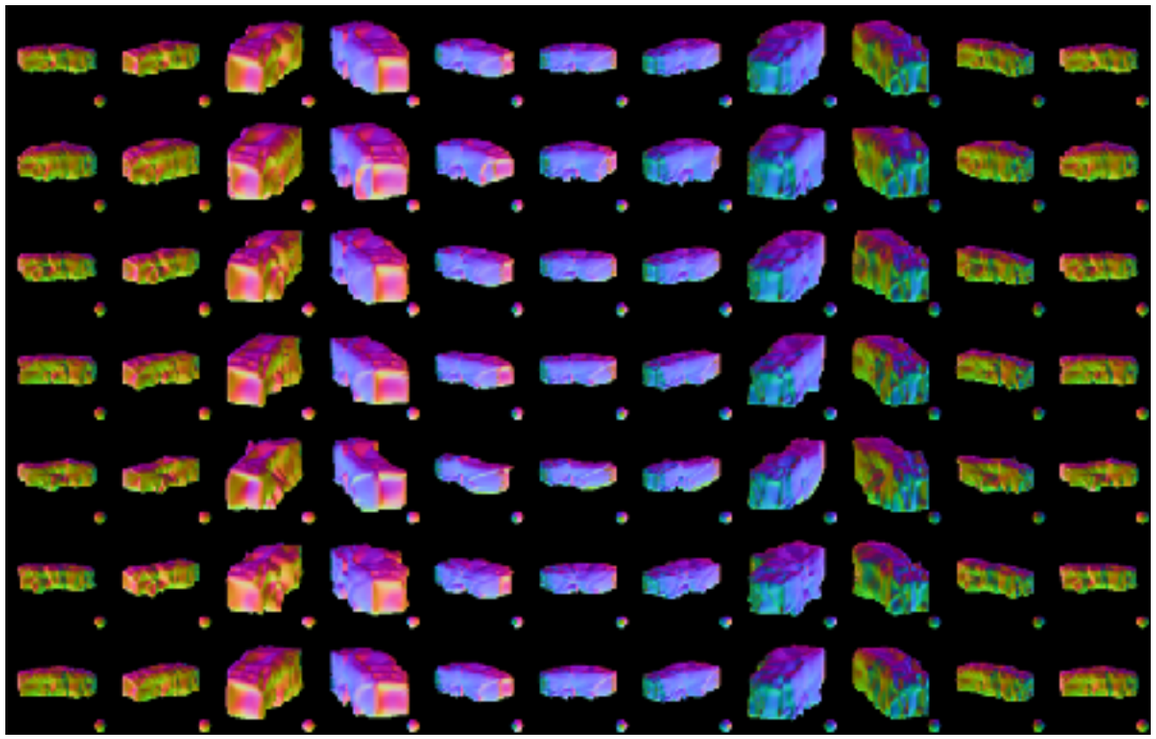


GAN on ShapeNet - Car



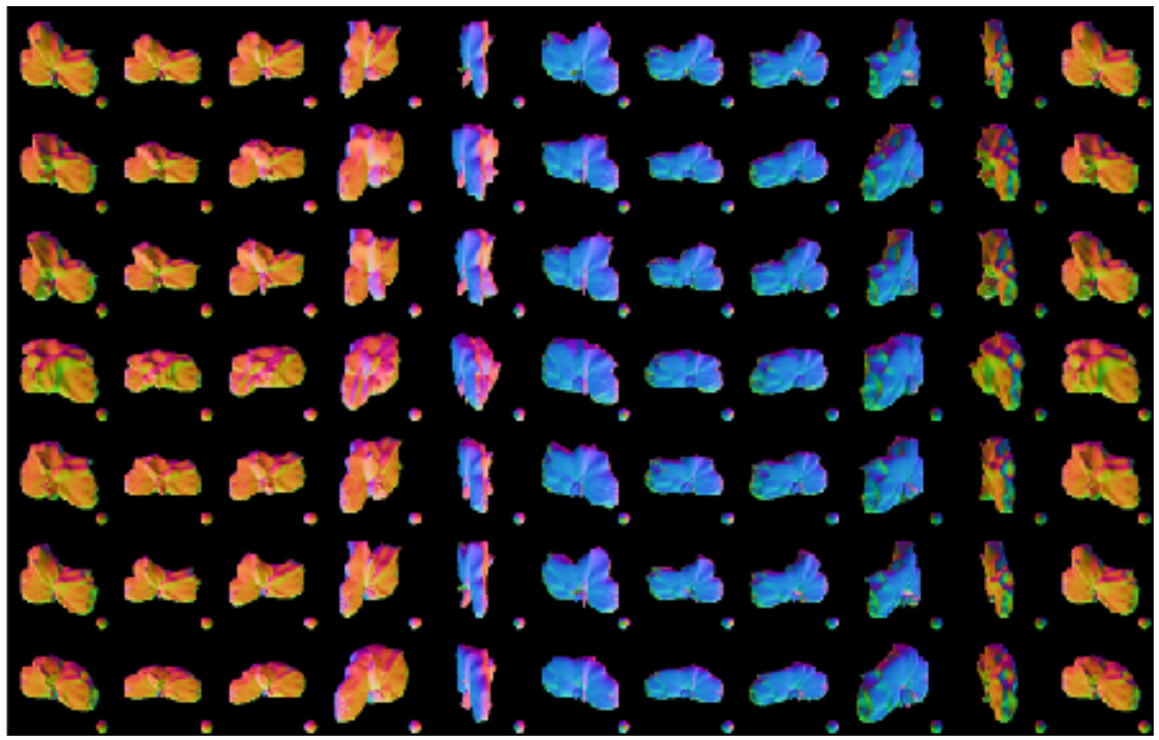
GAN on ShapeNet - Bus

GAN on ShapeNet - Bus

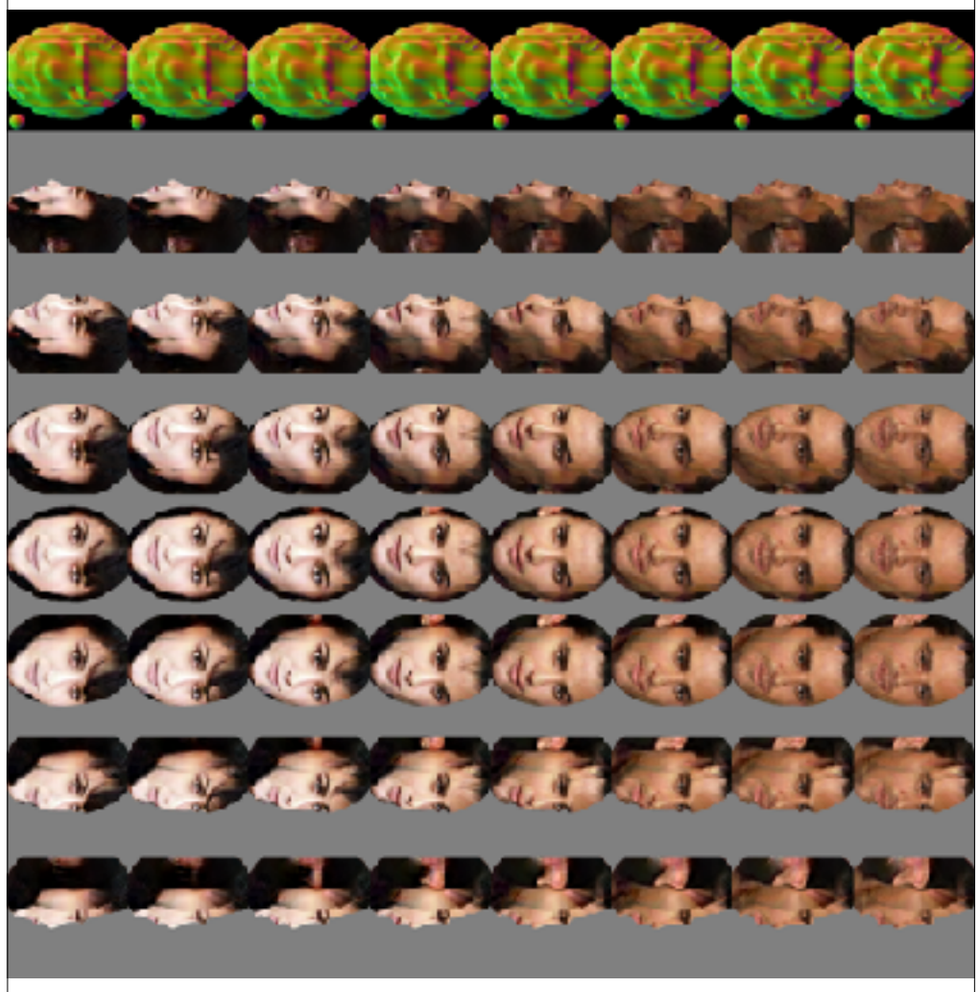


GAN on ShapeNet - Bike

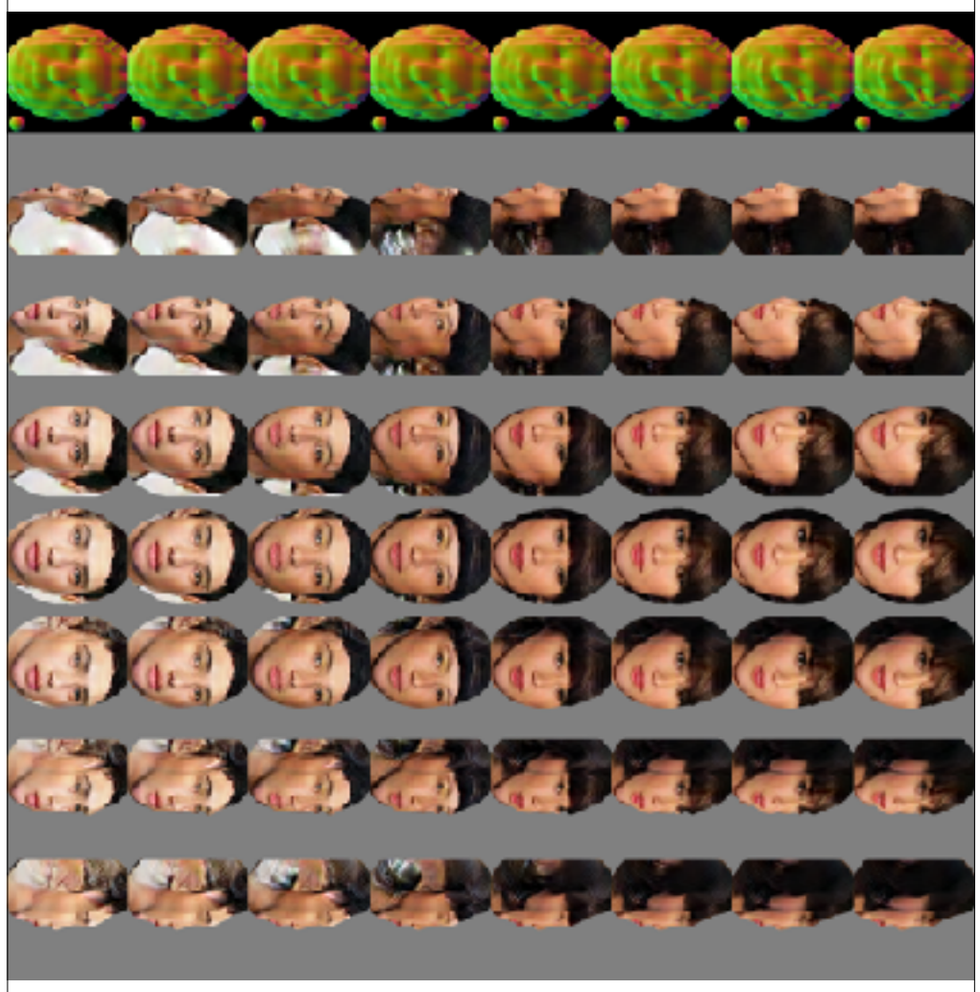
GAN on ShapeNet - Bike



interpolation



interpolation



Conclusions

- Self-supervised learning is a promising direction towards learning without human annotation
- Pretext-tasks can be defined
  - Directly on raw data
  - On known transformations of the data
  - On learned transformations of the data
- Evaluating features remains an open issue

Conclusions

- Unsupervised learning of factors of variation provides a different perspective to representation learning
  - An information-lossless representation is learned by ensuring that it leads to reconstruction
  - No annotation is needed

Conclusions

- Promising results on the disentanglement of
  - Viewpoints/car modes
  - Unknown sets of global attributes
  - 3D from texture, viewpoint and background
- Techniques (GANs, reconstruction constraints, differentiable renderers) are now quite effective
- Expect that generative models will become prominent

References

[1] M. F. Mathieu, J. J. Zhao, J. Zhao, A. Ramesh, P. Sprech- mann, and Y. LeCun. Disentangling factors of variation in deep representation using adversarial training. NIPS 2016.

[2] Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., and Abbeel, P. InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. ArXiv e-prints, June 2016.

[3] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. ICLR, 2016.

[4] E. Denton and V. Birodkar. Unsupervised learning of disentangled representations from video. NIPS, 2017.

[5] Z. Shu, E. Yumer, S. Hadap, K. Sunkavalli, E. Shechtman, and D. Samaras. Neural face editing with intrinsic image disentangling. CVPR 2017.

[6] L. Tran, X. Yin, and X. Liu. Disentangled representation learning gan for pose-invariant face recognition. CVPR 2017.

[7] X. Peng, X. Yu, K. Sohn, D. N. Metaxas, and M. Chandraker. Reconstruction-based disentanglement for pose-invariant face recognition. ICCV 2017.

[8] L. Ma, Q. Sun, S. Georgoulis, L. V. Gool, B. Schiele, and M. Fritz. Disentangled person image generation. CVPR 2018

[9] M. Liu, T. Breuel, and J. Kautz. Unsupervised image-to-image translation networks. NIPS 2017.

[10] A. Szabó, Q. Hu, T. Portenier, M. Zwicker, P. Favaro, Challenges in Disentangling Independent Factors of Variation, ECCV 2018

[11] Q. Hu, A. Szabó, T. Portenier, M. Zwicker and P. Favaro, Disentangling Factors of Variation by Mixing Them, CVPR 2018