

[논문 리뷰] Semantic Image Synthesis with Spatially-Adaptive Normalization

kimkj38 · 2021년 10월 31일

❤ 0

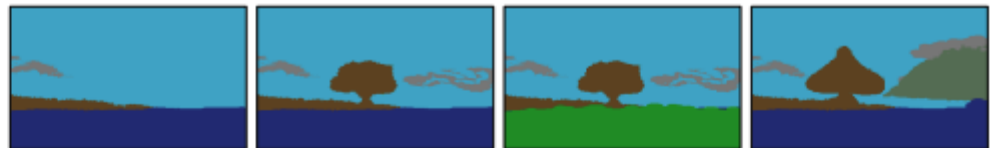
Paper gan

논문

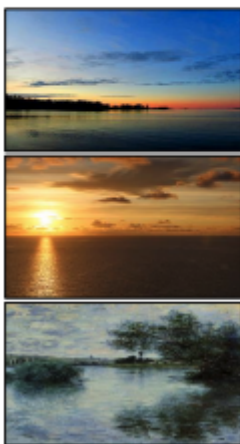


▼ 목록 보기

16/19



Semantic Manipulation Using Segmentation Map →



Style Manipulation using Style Images ↓



Introduction

- 본 논문에서는 **semantic segmentation mask**를 사진처럼 변환해주는 조건부 이미지 합성 방식을 제안한다.

- 기존에도 이와 같은 연구는 있었으나 semantic mask에 대해 **wash away 현상**이 나타난다는 문제점이 있었다.
- 따라서, **spatially-adaptive normalization** 기법을 활용하여 semantic 정보에서도 최적의 결과를 보일 수 있는 네트워크를 보인다.
- **semantic**과 **style**을 모두 **control** 할 수 있다.

Normalization

Unconditional Normalization

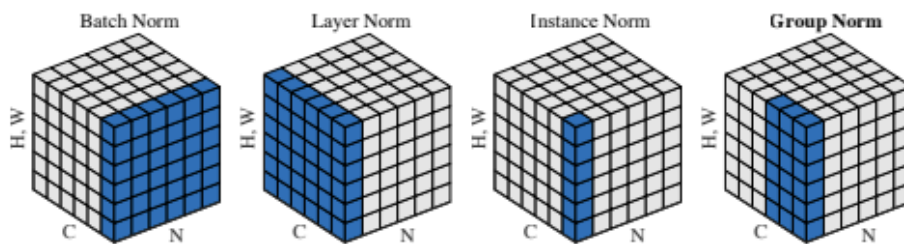


Figure 2. **Normalization methods.** Each subplot shows a feature map tensor, with N as the batch axis, C as the channel axis, and (H, W) as the spatial axes. The pixels in blue are normalized by the same mean and variance, computed by aggregating the values of these pixels.

Normalization의 형태는 $\gamma \frac{x - \mu(x)}{\sigma(x)} + \beta$ 로 동일하나 평균과 표준 편차를 구하는 방식에 따라 구분이 된다.

Batch normalization

N, H, W 에 대해서만 연산을 하여 배치별 normalization을 한다.

Layer normalization

배치와 무관하게 C, H, W 에 대해 연산하여 채널별 normalization을 한다.

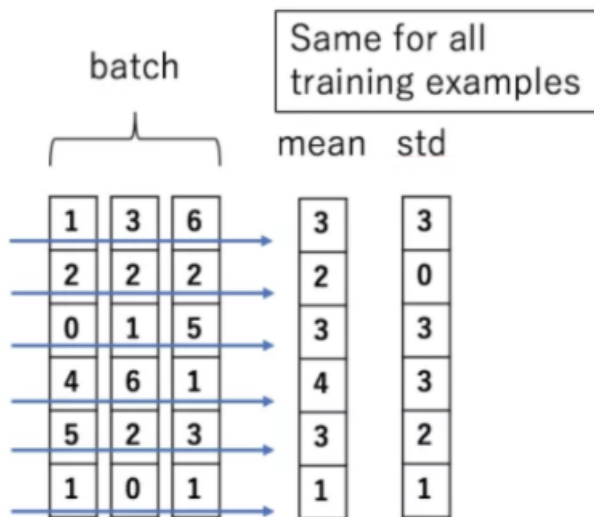
Instance normalization

$(1, H, W)$ 에 해당하는 부분으로 배치, 채널과 무관하게 각 데이터에 대해 normalization을 한다.

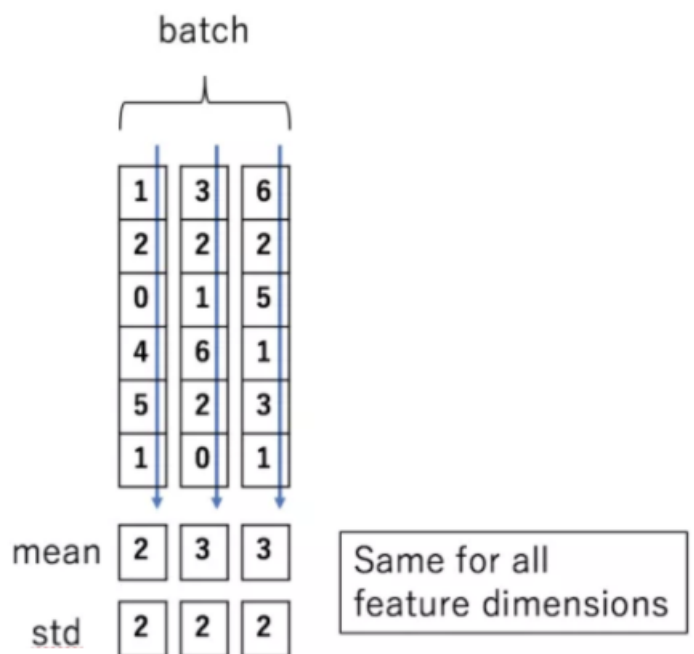
Group Normalization

채널을 그룹으로 묶어 normalization을 한다. 모든 채널을 한 그룹으로 묶이면 layer normalization과 동일하며, 하나의 채널을 한 그룹으로 보면 instance normalization과 동일하다.

Batch Normalization

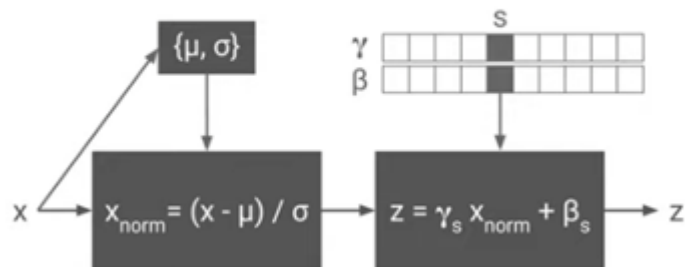


Layer Normalization

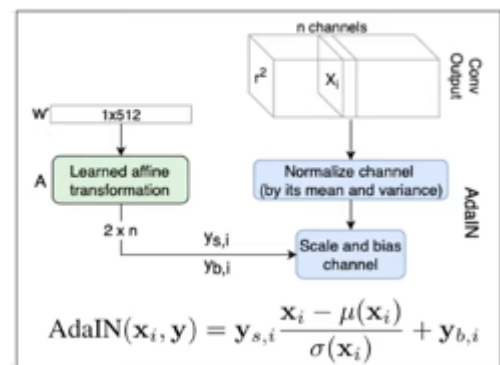


Conditional Normalization

Conditional Batch Normalization

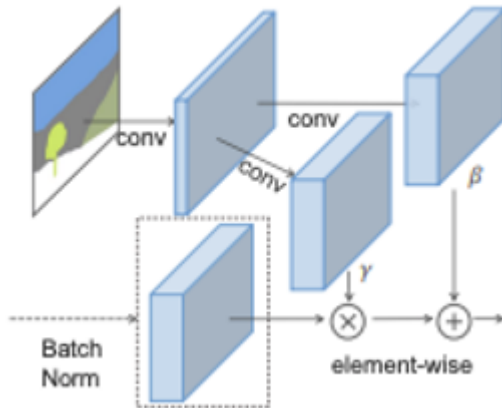


Adaptive Instance Normalization



- Normalization을 한 후 벡터 형태로 담겨있는 조건 변수를 곱하고(**scaling**) 더해준다(**bias**).
- AdaIN에서는 위와 같은 과정을 채널별로 수행한다.

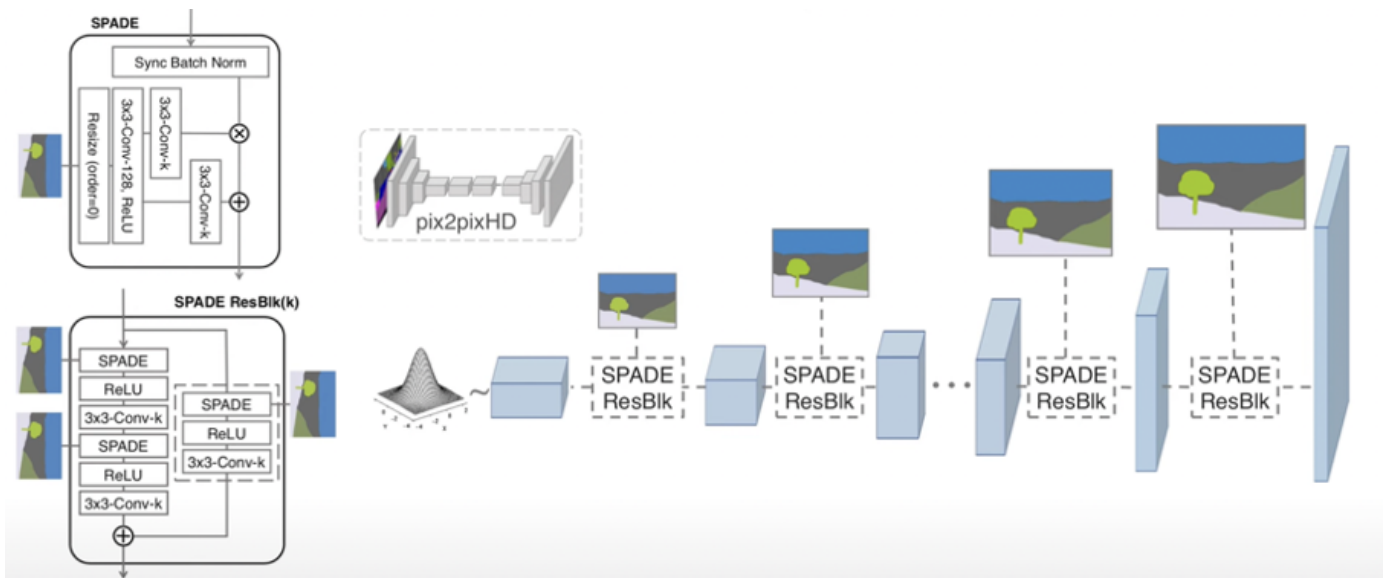
SPADE(Spatially-Adaptive Denormalization)



$$\text{external data } \gamma_{c,y,x}^i(\mathbf{m}) \frac{h_{n,c,y,x}^i - \mu_c^i}{\sigma_c^i} + \text{external data } \beta_{c,y,x}^i(\mathbf{m})$$

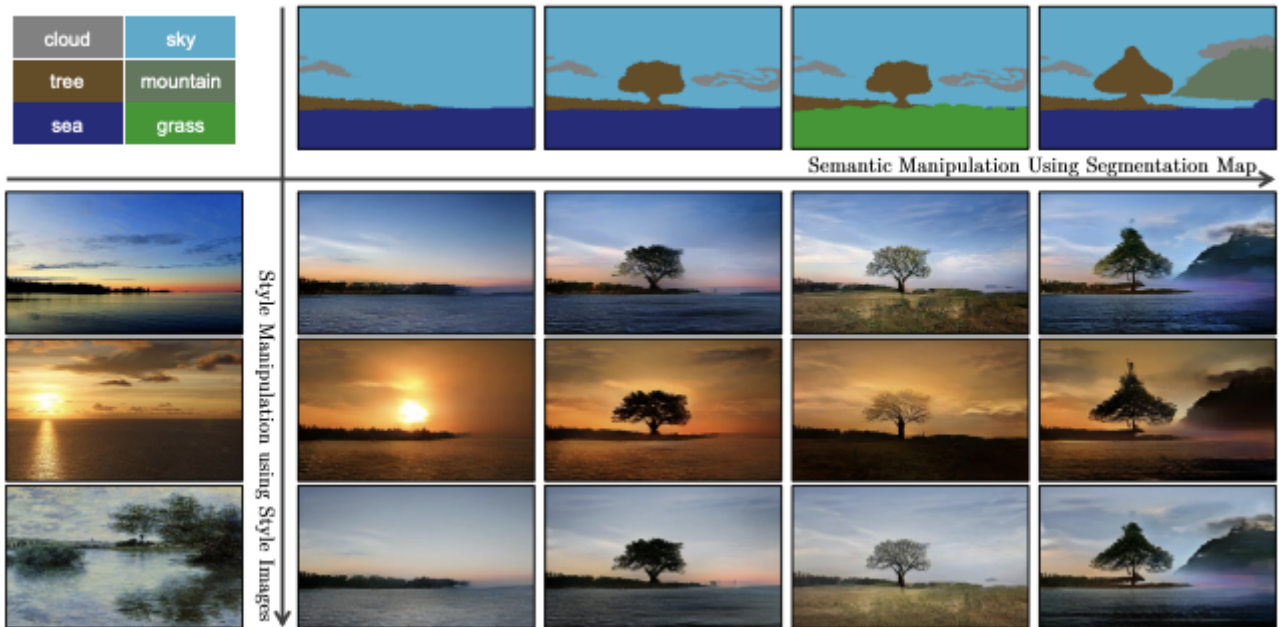
normalization

- 조건으로 주는 γ, β 를 본 논문에서는 **external data**라고 칭하며 **segmentation map**으로부터 뽑는다.
- 벡터 형태였던 기존 conditional normalization과 달리 convolution layer를 거쳐 **텐서** 형태로 나타나 각 픽셀별로 적용될 수 있다.



- SPADE는 **residual block** 형태로 구성된다.
- Learned modulation parameter(γ, β)에 충분한 정보가 인코딩 되어있으므로 첫번째 레이어에서 segmentation map을 input으로 받을 필요가 없으며 **encoder** 구조가 생략된다.
- Input으로 랜덤 벡터를 받아 **multi-modal synthesis**가 가능해졌다.
- Loss function은 pix2pixHD와 동일하며, least squared loss가 hinge loss로만 바뀌었다.

- 이미지로부터 평균과 분산을 뽑는 인코더를 만들면 이로부터 확률분포를 만들어 input값으로 활용할 수 있으므로 **이미지에 따라 스타일에 변화를 줄 수 있다.**
- 이 과정에서 VAE 구조를 가지기 때문에 **KL-Divergence Loss**를 추가하였다.
- Generator의 결과는 실제 이미지와 **concatenation**을 수행한 후 Discriminator로 전달된다.



References

- <https://arxiv.org/pdf/1903.07291.pdf>
- <https://arxiv.org/pdf/1803.08494.pdf>
- <https://www.youtube.com/watch?v=1nJf35TSYtE>
- <https://wingnim.tistory.com/92>
- <https://wdprogrammer.tistory.com/67>



김경준

코딩하는 문과생

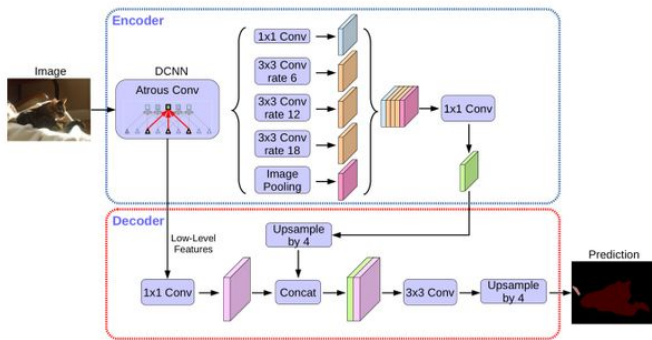
다음 포스트





이전 포스트

관심 있을 만한 포스트



DeepLab v3+ 논문 리뷰



Disentanglement



Disentanglement



Style GAN

StyleGAN(스타일갠)

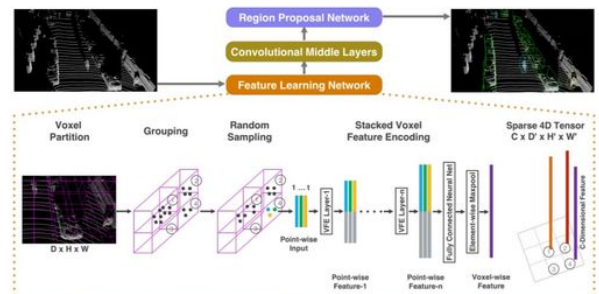


Figure 2. VoxelNet architecture. The feature learning network takes a raw point cloud as input, partitions the space into voxels, and transforms points within each voxel to a vector representation characterizing the shape information. The space is represented as a sparse 4D tensor. The convolutional middle layers process the 4D tensor to aggregate spatial context. Finally, a RPN generates the 3D detection results.

[Paper Review] VoxelNet: End-to-end Learning for Point Cloud Based 3D Object...

0개의 댓글

댓글을 작성하세요

댓글 작성