ECCV 2018
European Conference on Computer Vision
8 – 14 September 2018 | Munich, Germany

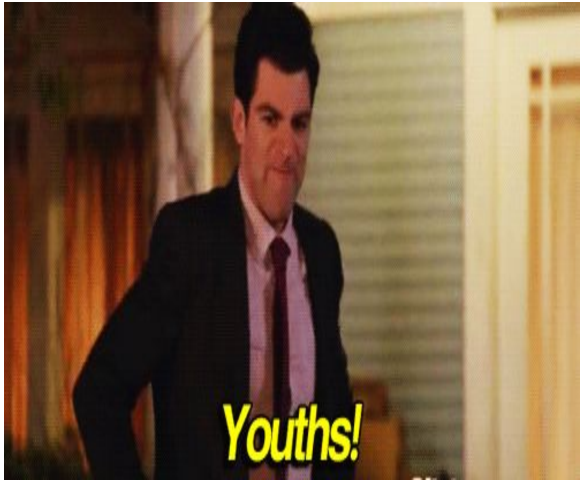
Video De-Captioning using U-Net with Stacked Dilated Convolutional Layers.

ChaLearn Video Decaptioning Challenge

Team :
Shivansh Mundra
Mehul Kumar Nirala
Sayan Sinha
Arnav Kumar Jain

Who are we?

Well, we are a bunch of undergraduates from India bonded together as a research community in Indian Institute of Technology, Kharagpur, India.



Let’s break down into steps

- Introduction
- Related Works
- Main Contribution
- Dataset
- Results
- Conclusion
- Future Work

Introduction

Aim: To develop algorithms to remove text overlays in video sequences

The problem of **Video De-Captioning** can be broken down into two phases:

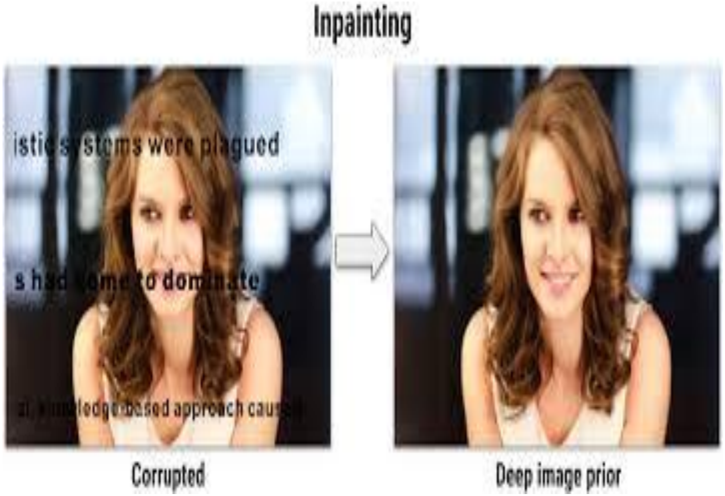
- **De-Captioning** of individual frames
- Processing the data as **continuous frames** of the videos

Related Works

- Video Inpainting by jointly learning temporal structure and spatial details.
    - Wang et al.
    - Main Contributions
        - Take mask as input.
        - Temporal structure inference by 3D Convolutional Networks.
        - Spatial details completion by Comb Convolutional Networks.
- Image Denoising and Inpainting with deep neural networks. (NIPS 2017)
    - Used stacked sparse denoising encoder-decoder architecture.
    - Images were of specific genre.
    - Dataset used for experimentation had gray scale images.

Why not use state-of-the art method for video/image inpainting?

- Video frames were not from a specific class/genre
- Trained on specific classes.
- Low resolution videos doesn't allow flexibility in exploring deep architectures.

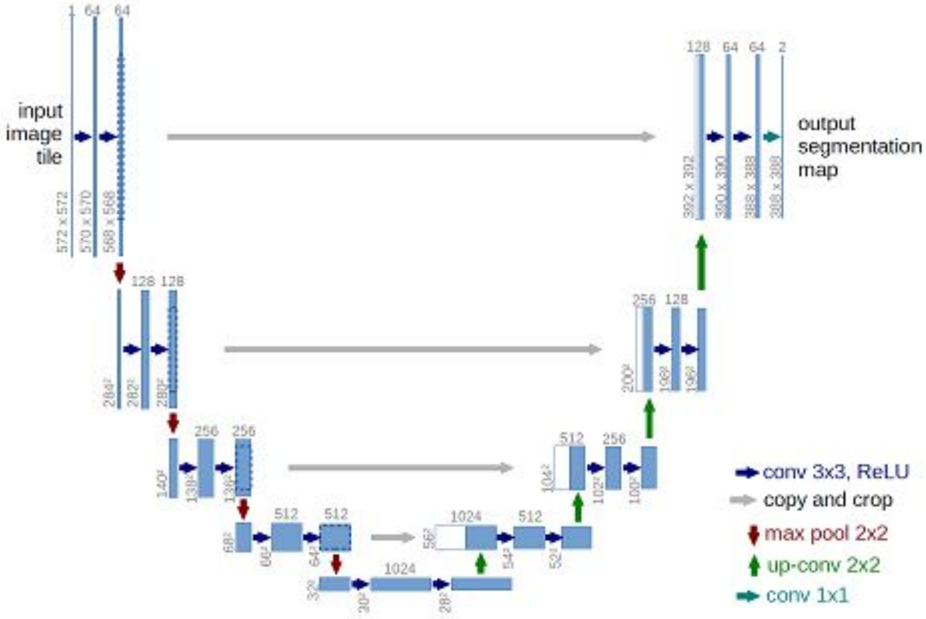


Main Contribution

- **U-Net** based encoder-decoder architecture
- Stacked **Dilated Convolutions layers** in encoder in the architecture
- Residual connections of convolutions in the bottle neck layer of encoder-decoder
- Converted all data to TFRecords for better performance

What is U-Net?

An encoder decoder based image segmentation model is used a lot for medical imaging, segmentation etc.



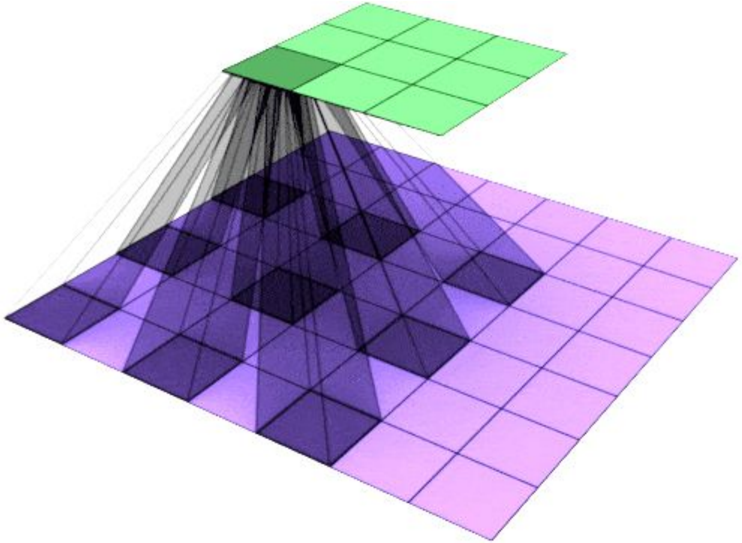
Features of U-Net Architecture

- Encoding with 3x3 kernel (no padding) followed by ReLu units
- Decoding part with 2x2 deconvolution at a time
- Concatenation of symmetrical layers in encoder-decoder

Stacked dilated Convolutional Layers

- Dilated convolutions introduce another parameter called the **dilation rate**
- Defines spacing between the values in a kernel
- A 3x3 kernel with a dilation rate of 2 will have the same field of view as a 5x5 kernel, while only using 9 parameters
- Imagine taking a 5x5 kernel and deleting every second column and row

Generative Image Inpainting with Contextual Attention Yu et Al



Why Stacked dilated Convolutional Layers ?

- Discrete Convolutions gives output of adjacent pixel space.
- Dilations increase the total receptive field
- Dilated convolutions are especially promising for image analysis tasks requiring detailed understanding of the scene
- Dilated Convolutions avoids needs of upsampling
- This delivers a wider field of view at the same computational cost


about this what took
I care about you
but I never thought
of you that way you

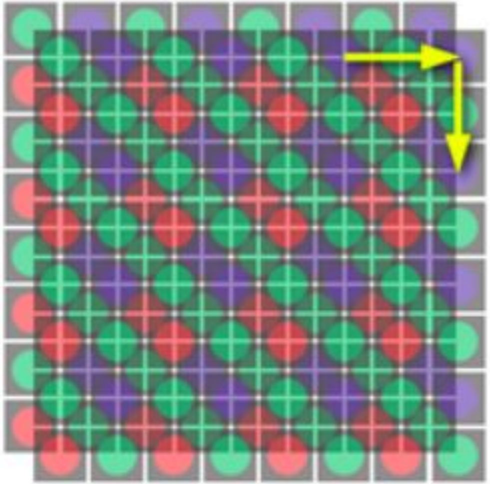
Residual Connections in bottle neck layer

- Residual connections are helpful for simplifying a network's *optimization.*
- They are used to allow gradients to flow through a network directly, without passing through non-linear activation functions.

Loss functions

- We trained our model on MSE loss and regularized it by **Total Variation Loss** and **PSNR** loss.

Total Variation Loss -:



Prediction Pipeline

- For predicting test videos we used approach given in baseline
- Divide image into 16 equal squares
- Check whether a square contain text
- Replace with original if doesn’t contain text

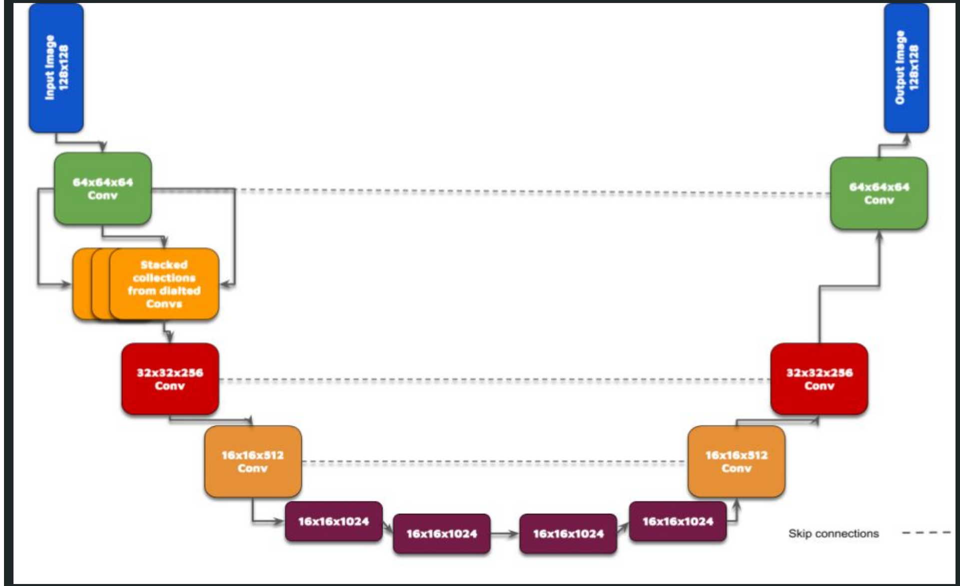
Features of Dataset

- Video duration : 5 sec
- Number of frames : 125
- Resolution of single frame : 128x128x3
- Train-val-test split :
  - Training - 10,000 videos
  - Val - 5,000 videos
  - Test - 5,1000 videos
- Videos were from diverse classes collected from Youtube
- Percentage of area covered from text was variable between 10%-60%

Results

Results	MSE	PSNR	DSSIM
Baseline	0.0022	30.1856	0.0613
Ours	**0.0012**	**32.1713**	**0.0482**

Average Execution time for converting single video - 5 sec



Our Solution Architecture

Rank	User	<Rank>	MSE	PSNR	DSSIM
None	KAIST-RCV	2.6667	0.0011	33.3527	0.0404
None	ucs	3.3333	0.0011	33.0052	0.041
None	hcilab	3.0	0.0012	33.0228	0.0424
None	anubhap93	3.6667	0.0012	32.0021	0.0499
None	arnavkj95	4.0	0.0012	32.1713	0.0482
None	Baseline	4.3333	0.0022	30.1856	0.0613

The problem of **De-Captioning**

The problem of De-Captioning was different from the usual problem of **inpainting** :

- Position and orientation of subtitles was specified(in center bottom)
- Inpainting involves filling a whole region/patch
- De-Captioning involves inpainting of regions which are covered by texts.

Conclusions

- Encoder-Decoder network can be used for inpainting/decaptioning
- Our solution doesn’t require **mask** as input hence we were able to decrease computation time
- The proposed solution can be applied to any class of video-to-video or image-to-image translation in very less execution time
- Old GANs approaches weren’t able to generalise well in the dataset from domains.

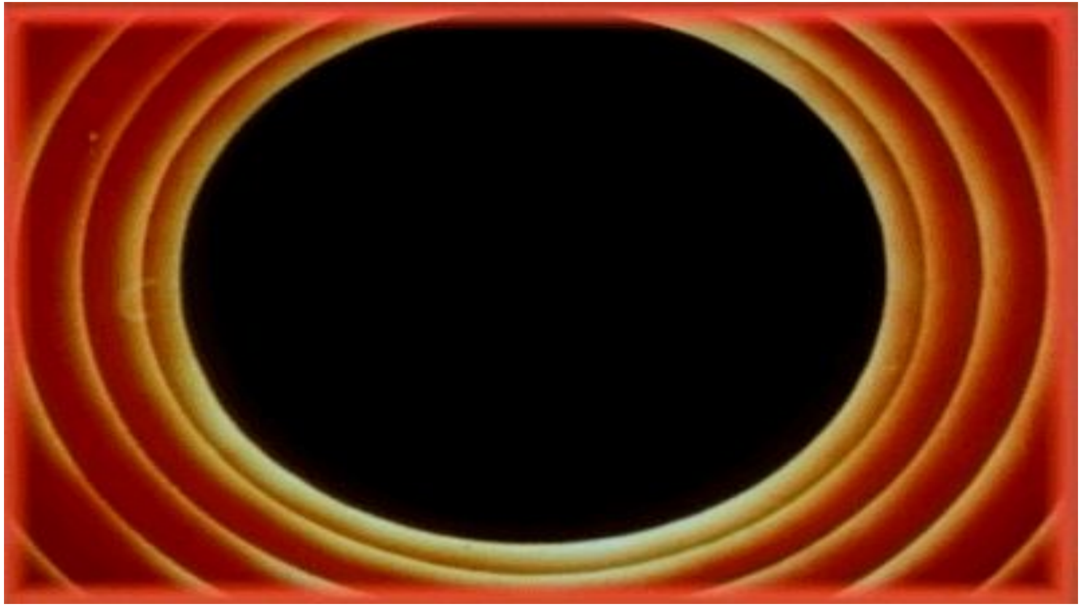
Conclusions...

- We tried regularizing our model with VGG feature loss which resulted in more appealing videos but MSE error increased

Future Works

- Exploiting Temporal relations in Videos
  - Temporal context and a partial glimpse of the future, allow us to better evaluate the quality of a model's predictions objectively.
  - Can take advantage of the frames in stack which don’t have subtitles
  - 3D Convs can extract temporal dimension with motion compensation.
- Diverging from end-to-end learning
  - Training first to predict mask, then inpaint corresponding mask.

That’s All



Thanks!

Indian Institute of Technology Kharagpur.