# Convolutional Neural Networks for Direct Text Deblurring (Hradiš et al)

Instructor: Dr. Fabrice MERIAUDEAU

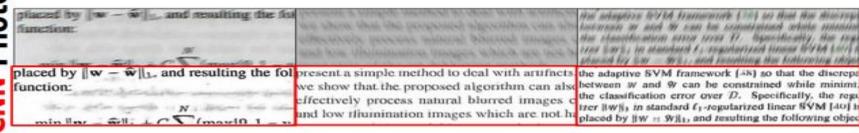
Jimoh Fatai Olarinde (<u>Fatai\_jimoh@etu.u-bourgogne.fr</u>)
Muhammad Roshan Mughees (<u>Muhammad-Roshan\_Mughees@etu.u-bourgogne.fr</u>)
Syed Nouman Hasany (<u>Syed-Nouman\_Hasany@etu.u-bourgogne.fr</u>)





## Contents

- Problem Statement
- Overall Pipeline
- Dataset acquisition and details
- Convolutional Neural Networks
- Proposed Architecture
- Results and Conclusion
- Future Work
- Demo



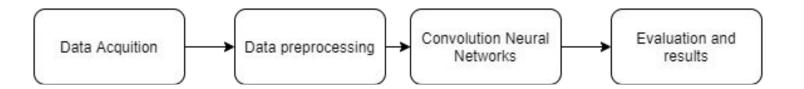
#### **Problem Statement**

- Blurring is a common phenomena
- Blurring can cause loss of information
- Can occur due to various reasons such as defocusing and camera shake

Convolutional Neural Networks have proven very powerful for Computer Vision problems in recent years.

We use a CNN and train it for the restoration of **good-quality text images** directly from blurry inputs without assuming any **specific blur** or **noise models** 

# Overall Pipeline



#### **Dataset Details**

- Patches sampled from documents downloaded from CiteSeerX repository (available online on HRADIŠ' website: <a href="https://www.fit.vutbr.cz/~ihradis/CNN-Deblur/">https://www.fit.vutbr.cz/~ihradis/CNN-Deblur/</a>)
- Mix of different content types (text, equations, tables, images, and diagrams)
- 50k files for training and 2k files for validation
- Original Patch Dimensions: (300 x 300 x 3)
- Randomly sampled a patch from each patch
- Our Dimensions: (64 x 64 x 3)
  - Reason: Computation cost
- Normalized all images from 0 to 1

# Convolutional Neural Networks - Rationale

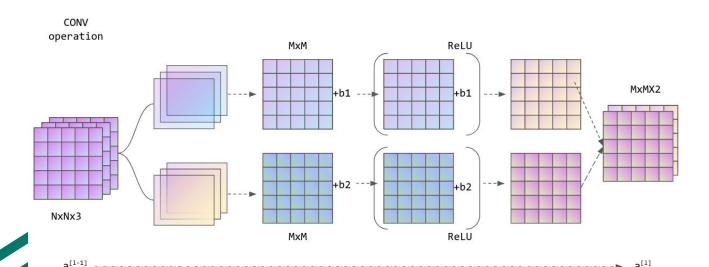
• Filters can provide us with useful information from the image





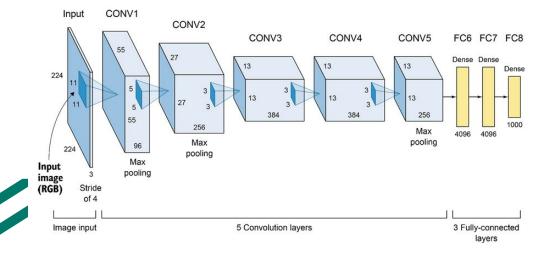
# Convolutional Neural Networks - Rationale

- Idea: Use multiple filters to extract different 'kind' of useful information about the image
- Learn these filters automatically to suit our desired task => CNNs



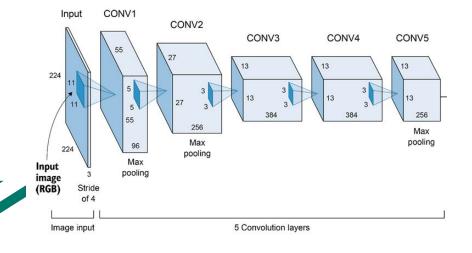
# Convolutional Neural Networks - AlexNet

- Winner of the 2012 ImageNet challenge => kick-started the Deep Learning revolution in Computer Vision
- Conv Pool Conv Pool Conv Conv Conv Pool Flatten Dense Dense Output



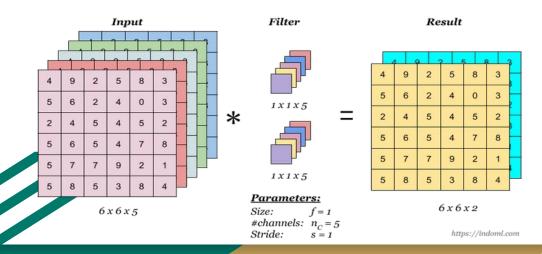
# Convolutional Neural Networks - Task at Hand

- Our task is not related to image classification => we want an image at the output
- Idea: Use a fully convolutional network with no dense layers



# Convolutional Neural Networks - 1x1 Convolutions

- 1x1 Convolutions are often used to change the number of channels without using as many parameters as a conventional filter (3x3, 5x5, etc.)
- They can also be thought of as applying intensity transforms (without any consideration for the neighbours)



#### Our Architecture

- Architecture proposed by the authors
  - o L-15:

- Consists of successive convolutional layers
- Ends up having the same dimension as the image

	1														
T 15	19×19 128	$1 \times 1$	$1\times1$	$1 \times 1$	$1\times1$	$3\times3$	$1 \times 1$	$5\times5$	$5\times5$	$3\times3$	$5 \times 5$	$5 \times 5$	$1 \times 1$	7×7	7×7
LIS	128	320	320	320	128	128	512	128	128	128	128	128	256	64	3

# Our Architecture (Contd.)

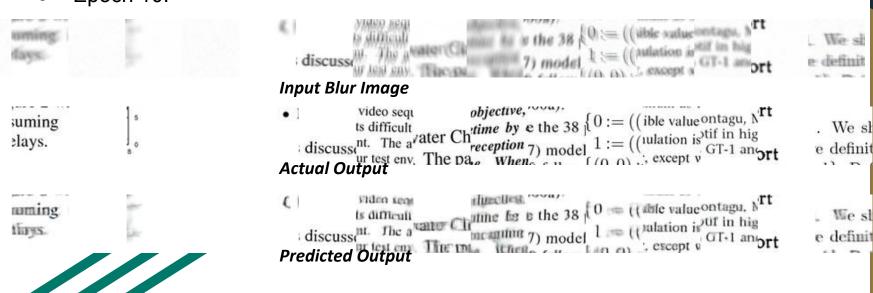
- Training parameters:
  - 15 Layers CNN model
  - Padding: same
  - No. of Epochs: 20
  - Avg. time per epoch: 54 mins 12 sec
  - o Batch size: 16
  - Optimizer: Adam
  - Activation Functions: ReLU, Sigmoid => last layer)
  - Loss: Mean squared loss (according to the paper)

- Results per 5 Epoch intervals on validation set:
- Epoch 5:

the I	rank-less, mu T evolutio	he case	TI
New	Input Blur Image	(malada)	tio
the I	rank-less, mues in e305.2 ion 3.2.) circled dcan be lirectly n	ite, $t_1$ hahe case error to ar	Th
New	separa orics, creed with tlt. mayorinter w	e probabi	tio
the l	rank less, mues in case i ion M cucledd an he inner n	in, a hahe case	Th
Now	T evolution icics creed with a Predicted Output	Madama	tio

he case	The tion	ent inse
ite, t <sub>1</sub> hahe case error to ar	iefly The tion	nes [Ba   x   x   x   nent inse
in, a hine case morte ar	iefly The tion	nes [Ba lent inse lility of

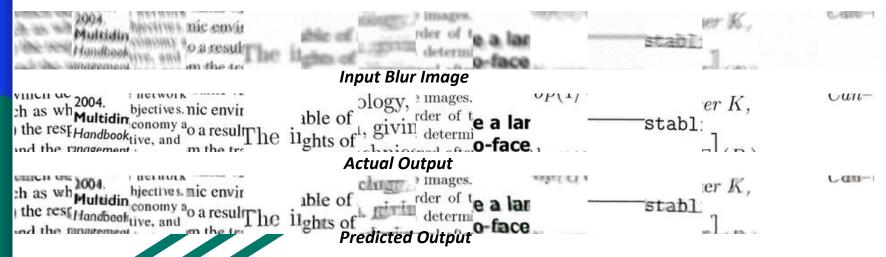
• Epoch 10:



Epoch 15:

e to help at of Mo ever, we Input Blur Image т паши  $-x_m$ sualizatithese tes 2mpressiv Sto plans thto 0, bar that I ally resist tof Mo S: e to help a very rsensitized eaches the from term a in equa, Grouioble ever, we l-shaped examirilistic err Attribut.48 **Actual Output** - Ten sudiganthese tes ı uanını 2mpressiv Sic plans theo 0, but that I ally resist of Mo e to help a recry (sensiting eaches thee from term a in ceptal, Growoble ever, we orsamtidhstie om shape **Predicted Output** 

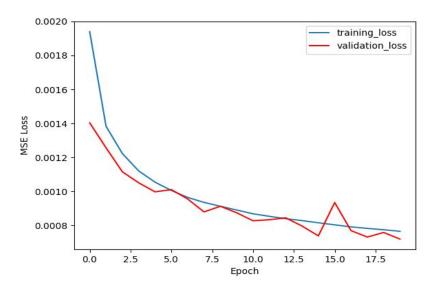
### • Epoch 20:



- From 64x64x3 (RGB input)
- To 64x64x3 (output)

Validation avg PSNR: 19.6 (final epoch)
Validation MSE: 0.000720 (final epoch)

- 0.001257 (first epoch)



Learning curve

#### **Future Directions**

- Training for more epochs
- Using recent architectures (U-Net, FCN)
- Regularization techniques (batch normalization, dropout)
- Hardware availability

#### References

[1].

https://www.semanticscholar.org/paper/The-Analysis-and-Implementation-of-Edge-Detection-Yang-Jin/c9c233dc29dd3f3b10 a94e7fb0a0d28b2d47af08

[2] https://github.com/dair-ai/ml-visuals

- [3] https://indoml.com/2018/03/07/student-notes-convolutional-neural-networks-cnn-introduction/
- [4] https://livebook.manning.com/book/grokking-deep-learning-for-computer-vision/chapter-5/v-3/
- [5] https://www.fit.vut.cz/research/publication-file/10922/hradis15CNNdeblurring.pdf

# Thank you for the opportunity professor !! We learnt a lot :)

**Any Questions?** 



