Deep Learning

Machine Learning

Prof. Sandra Avila
Institute of Computing (IC/Unicamp)

MC886, October 9, 2019

Deep Learning:
Hype or Reality?

Today's Agenda

- What is Deep Learning?
- Deep Learning & Applications

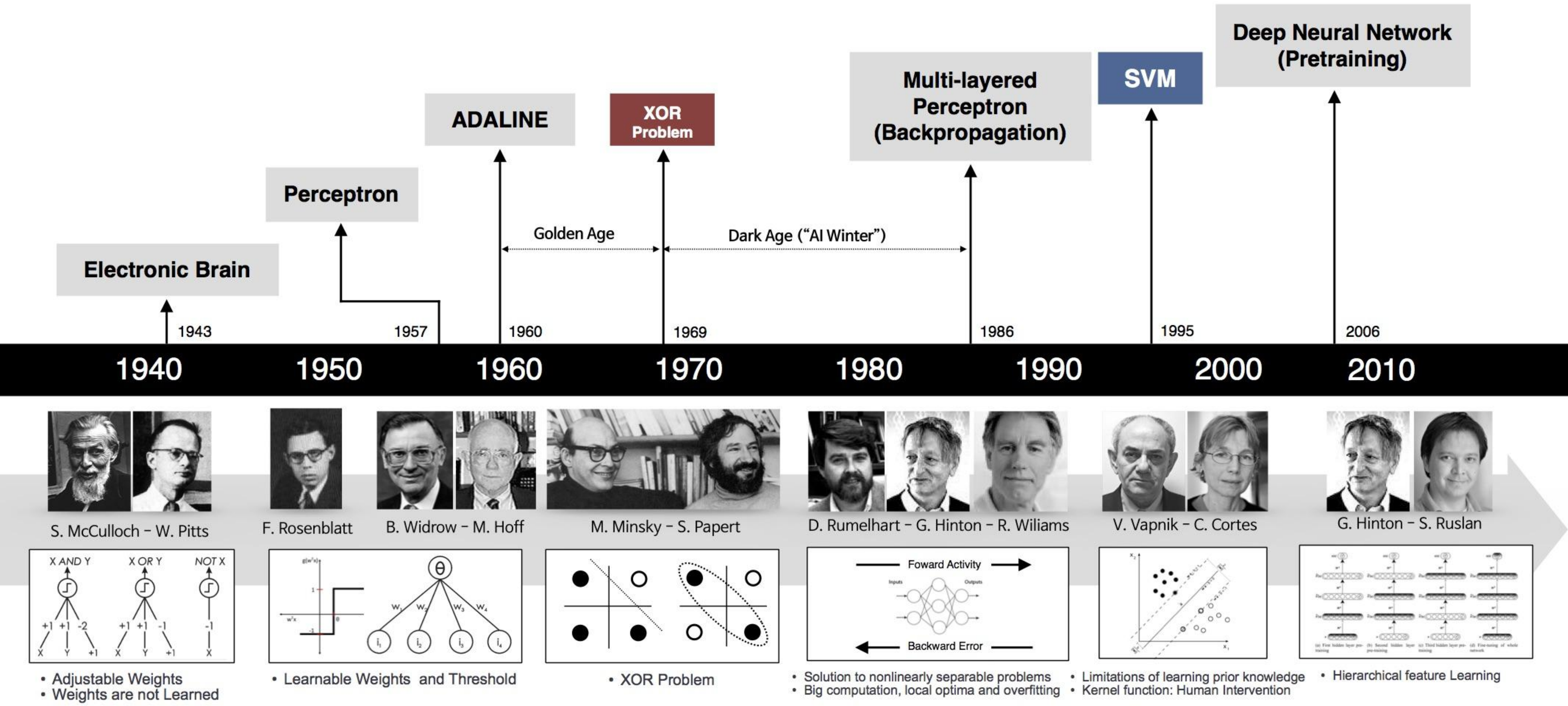
What is
Deep Learning?

“Deep learning allows computers to learn from experience and understand the world in terms of a hierarchy of concepts, with each concept defined through its relation to simpler concepts.

[Goodfellow & Bengio & Courville, 2016]

"Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction."

[LeCun & Bengio & Hinton, 2015]



Note: I am not the creator of the above image. I have been unable to locate the original source. If this is your image please let me know so I can give you proper attribution.

Traditional Recognition

Classifier

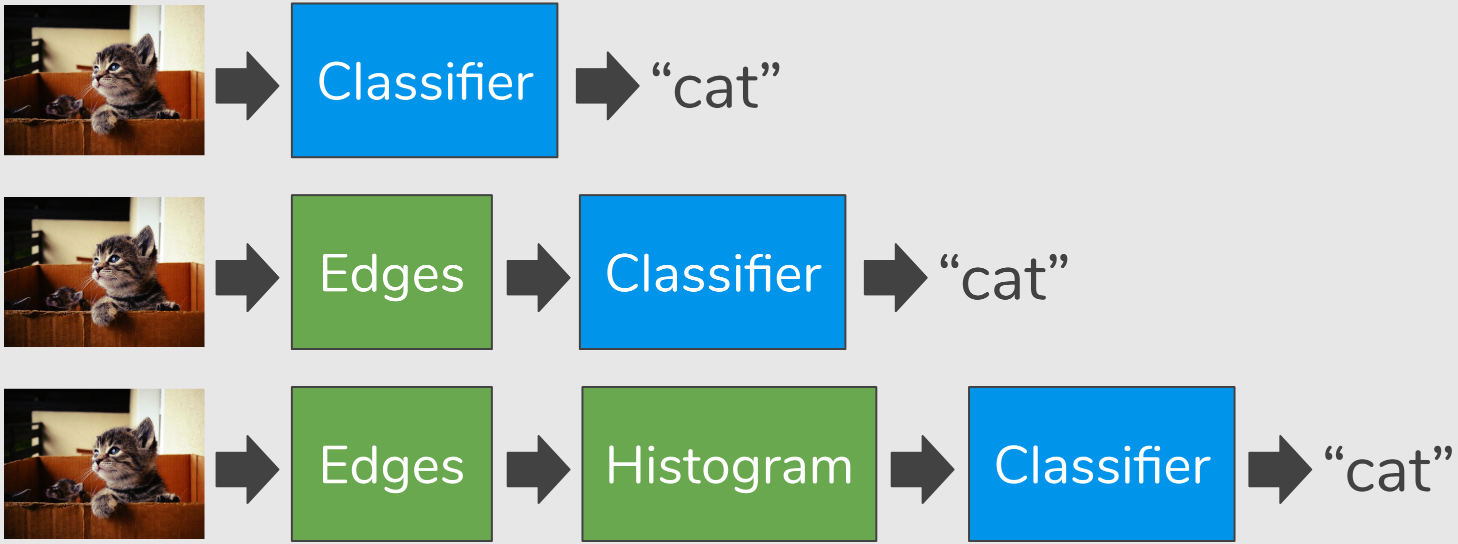
"cat"

Traditional Recognition

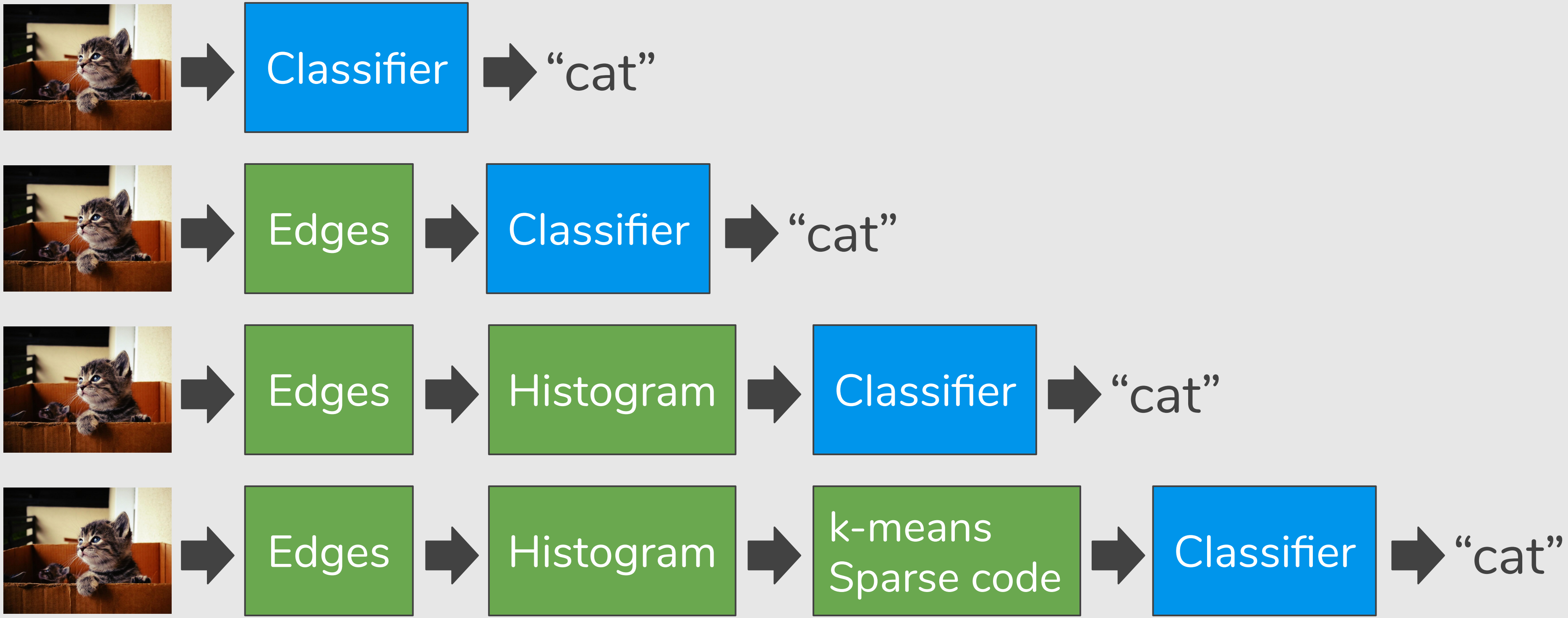
 → Classifier → “cat”

 → Edges → Classifier → “cat”

Traditional Recognition



Traditional Recognition



Deep Learning

Specialized components

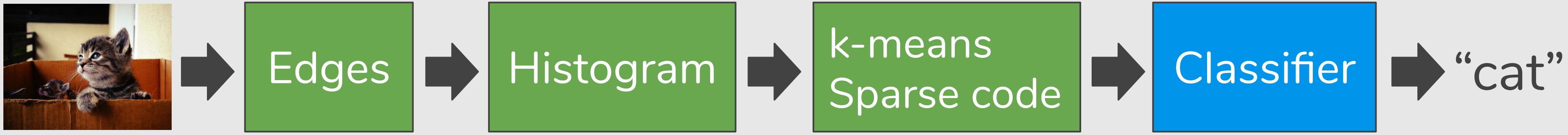
→ Edges → Histogram → k-means Sparse code → Classifier → “cat”

Generic components

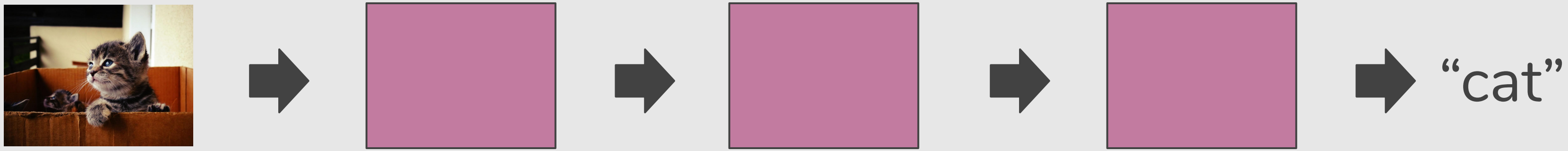
→ [ ] → [ ] → [ ] → “cat”

Deep Learning

Specialized components



Generic components



Generic components, going deeper

Deep Learning: Applications

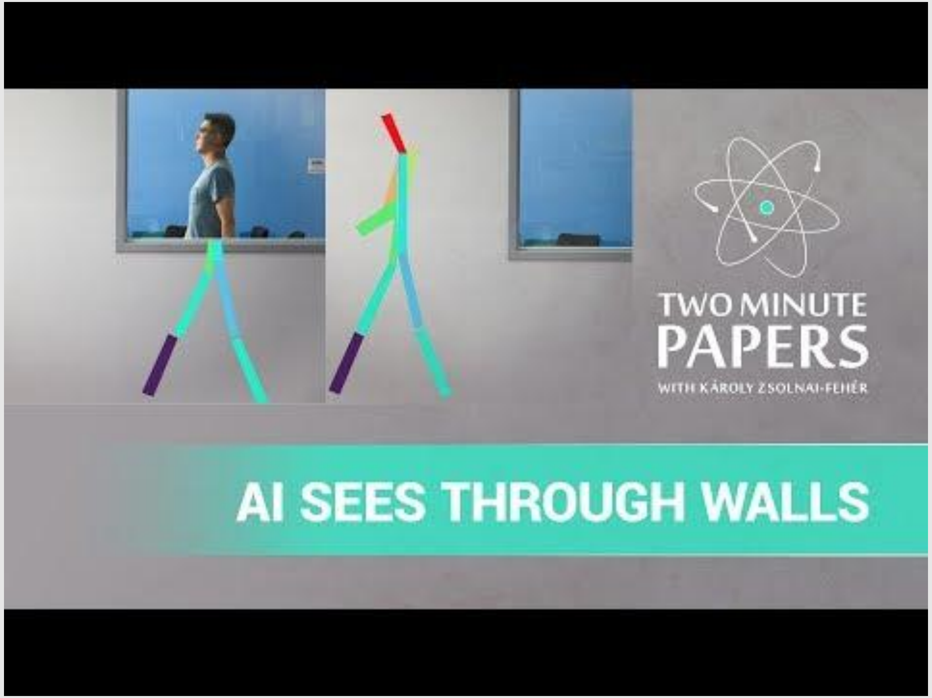
DL is everywhere ... pose estimation

“Realtime Multi-Person 2D Human Pose Estimation using Part Affinity Fields”, CVPR 2017

KTH Dataset (2005)

http://www.nada.kth.se/cvap/actions

DL is everywhere ... pose estimation



"Through-Wall Human Pose Estimation Using Radio Signals", CVPR 2018

DL is everywhere ... grape detection & segmentation

"Grape Detection, Segmentation and Tracking using Deep Neural Networks and Three-dimensional Association" (under review)

DL is everywhere ... Elsgate classification

“Combating the Elsagate Phenomenon: Deep Learning Architectures for Disturbing Cartoons”, IWBF 2019

Unicamp cria tecnologia para barrar pornografia e violência

Segurança. Pesquisadores lançaram método que identifica cerca de 97% do conteúdo impróprio em telas de celulares e computador

Em parceria com pesquisadores do Samsung Research Institute Brazil, o IC (Instituto de Computação) da Unicamp (Universidade Estadual de Campinas) desenvolveu um método capaz de filtrar 97% do conteúdo pornográfico e 80% do material de violência exibido em telas de celulares, computadores e tablets.

No novo método, os pesquisadores buscaram a combinação do uso de informações estáticas e de movimento com uma metodologia de aprendizado de máquina conhecida como deep learning ou "aprendizagem profunda". Com isso, a solução que o grupo desenvolveu extrai um quadro por segundo de cada vídeo que é acessado em tempo real em celular ou computador. Os quadros com as imagens estáticas são em seguida analisados aplicando-se o método de classificação de descrições do que é permitido e do que é pornográfico.

Ao mesmo tempo, a sequência de quadros analisados fornece os elementos para sequenciar os movimentos dos objetos e pessoas presentes na cena. Dependendo do tipo de movimento, o vídeo é bloqueado.

"Para a detecção de pornografia, os testes foram realizados em um conjunto de dados contendo aproximadamente 140 horas, sendo 1 mil vídeos pornográficos e 1 mil vídeos não pornográficos", explica a pesquisadora do IC da Unicamp, Sandra Avila, ao comentar sobre o processo de criação da tecnologia, que durou 27 meses.

Sistema garante proteção de crianças | IMAGE SOURCE/FOLHAPRESS

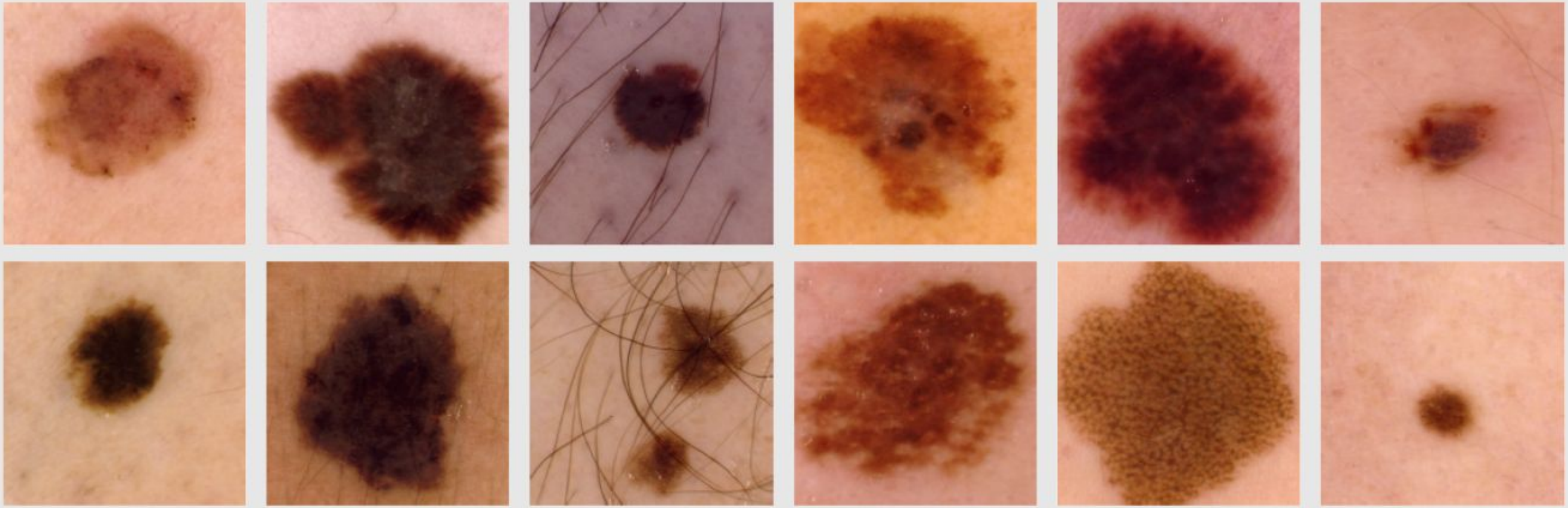
"Filtrar cenas de violência, por ser mais subjetivo, é um problema mais difícil comparado à pornografia. Devido a essa subjetividade e os diferentes conjuntos de dados, a eficácia da nossa solução para filtrar cenas de violência está em torno de 80%", conta Sandra.

Ainda segundo a representante da Unicamp, a tecnologia lançada em parceria com a Samsung pode ajudar as autoridades policiais.

"O método proposto para filtrar conteúdo pornográfico está sendo adaptado para outros tipos de conteúdo sensível. Por exemplo, em parceria com peritos da Polícia Federal, estamos desenvolvendo uma ferramenta para detectar pornografia infantil. Temos hoje uma solução que identifica 88% do conteúdo pornográfico infantil em imagens. Para dar uma ideia da importância do resultado, o melhor resultado alcançado pelas ferramentas forenses testadas foi 58%", relata a pesquisadora.

HIDAIANA ROSA
METRO CAMPINAS

DL is everywhere … skin cancer classification



Melanomas (top row) and benign skin lesions (bottom row)

“Towards Automated Melanoma Screening: Proper Computer Vision & Reliable Results”, 2016

ÁREA DE SAÚDE ▾ FACULDADES E INSTITUTOS ▾ INGRESSO ▾ INFORMAÇÕES PARA... ▾

› A UNIVERSIDADE › ENSINO › PESQUISA › RELAÇÕES COM A SOCIEDAD

| 23, MAR - 2017 | 09:00 | COMUNIDADE INTERNA

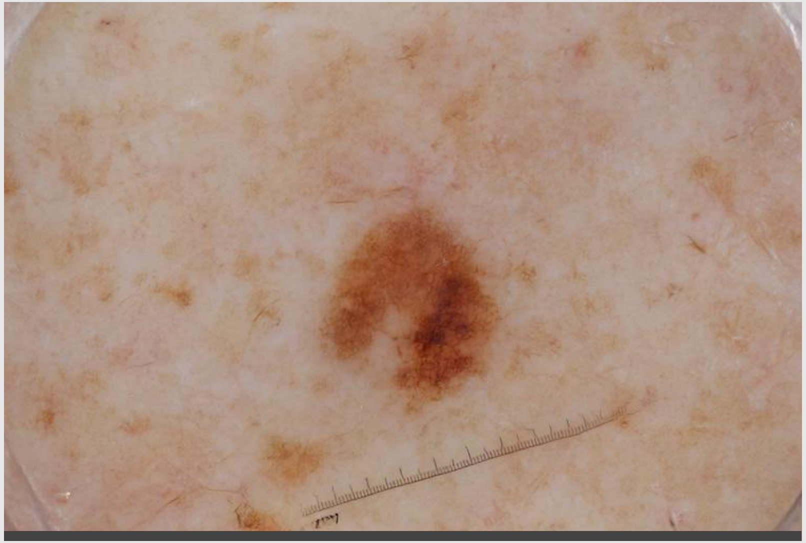
Equipe da Unicamp fica no topo de competição internacional de detecção automática de melanoma

| **Autor** Divulgação laboratório RECOD | **Fotos** Mijail Vidal | **Edição de imagem** Paulo Cavalheri

Uma equipe de professores e pesquisadores da Unicamp obteve excelente resultado na segunda edição da Competição Internacional de Análise de Lesões de Pele, evento anual não-presencial organizado pela Colaboração Internacional para Imagens de Lesões de Pele (ISIC). *Os* organizadores disponibilizam

https://www.unicamp.br/unicamp/noticias/2017/03/17/equipe-da-unicamp-fica-no-topo-de-competicao-internacional-de-deteccao

These images are not real!

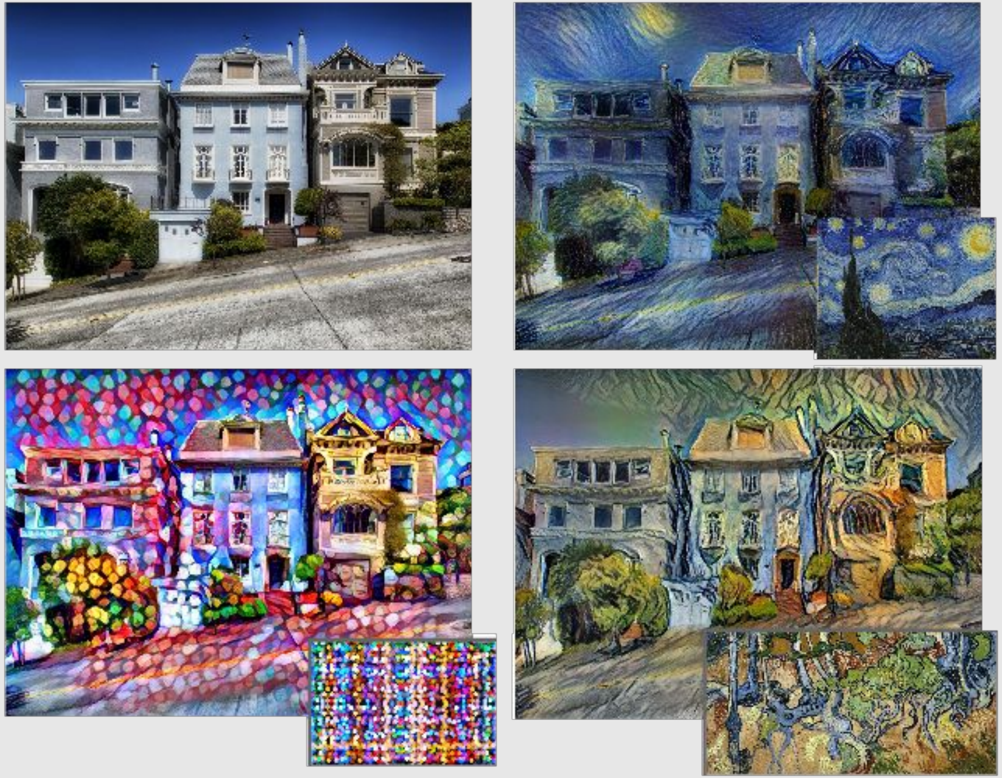


“Skin Lesion Synthesis with Generative Adversarial Networks”, MICCAI 2018

DL is everywhere ... face synthesis

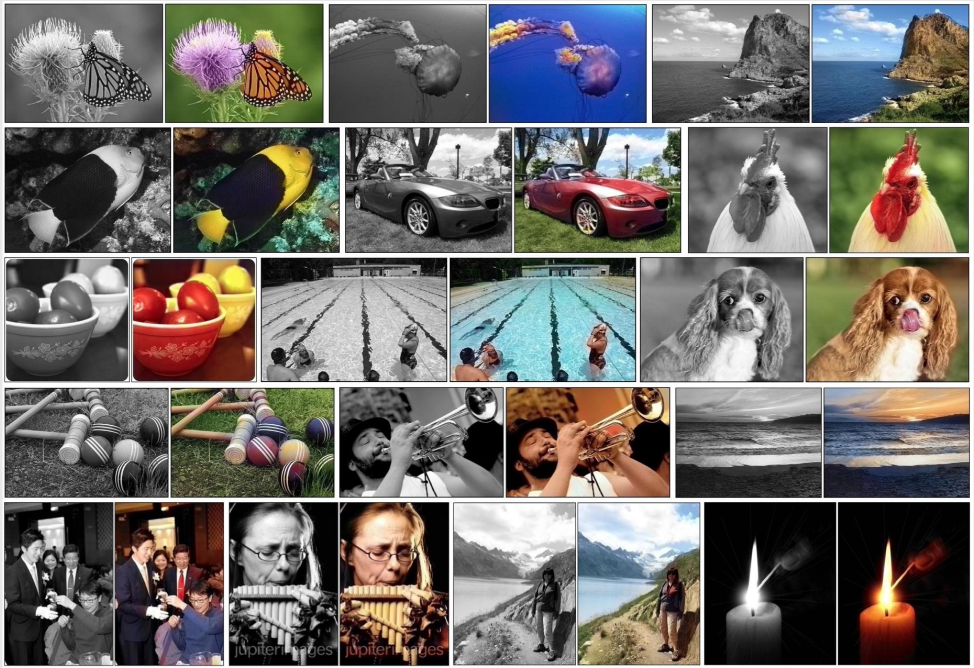
“A Style-Based Generator Architecture for Generative Adversarial Networks”, CVPR 2019

DL is everywhere … image style transfer



“Image Style Transfer using Convolutional Neural Networks”, CVPR 2016

DL is everywhere … image colorization



“Colorful Image Colorization”, ECCV 2016

DL is everywhere … audio synthesis

“Synthesizing Obama Learning Lip Sync from Audio”, SIGGRAPH, 2017

DL is everywhere … image captioning

Image Captioning

No errors

A white teddy bear sitting in the grass

A man riding a wave on top of a surfboard

Minor errors

A man in baseball uniform throwing a ball

A cat sitting on a suitcase on the floor

Somewhat related

A woman is holding a cat in her hand

A woman standing on a beach holding a surfboard

Captions generated by Justin Johnson using Neuraltalk.

DL is everywhere … text generation

Proof. Omitted. □

Lemma 0.1. *Let $\mathcal{C}$ be a set of the construction.*

Let $\mathcal{C}$ be a gerber covering. Let $\mathcal{F}$ be a quasi-coherent sheaves of $\mathcal{O}$-modules. We have to show that

$$\mathcal{O}_{\mathcal{O}_X} = \mathcal{O}_X(\mathcal{L})$$

.

Proof. This is an algebraic space with the composition of sheaves $\mathcal{F}$ on $X_{\acute{e}tale}$ we have

$$\mathcal{O}_X(\mathcal{F}) = \{morph_1 \times_{\mathcal{O}_X} (\mathcal{G}, \mathcal{F})\}$$

where $\mathcal{G}$ defines an isomorphism $\mathcal{F} \to \mathcal{F}$ of $\mathcal{O}$-modules. □

Lemma 0.2. *This is an integer $\mathcal{Z}$ is injective.*

Proof. See Spaces, Lemma ??. □

Lemma 0.3. *Let S be a scheme. Let X be a scheme and X is an affine open covering. Let $\mathcal{U} \subset \mathcal{X}$ be a canonical and locally of finite type. Let X be a scheme. Let X be a scheme which is equal to the formal complex.*

The following to the construction of the lemma follows.

Let X be a scheme. Let X be a scheme covering. Let

$$b : X \to Y' \to Y \to Y \to Y' \times_X Y \to X.$$

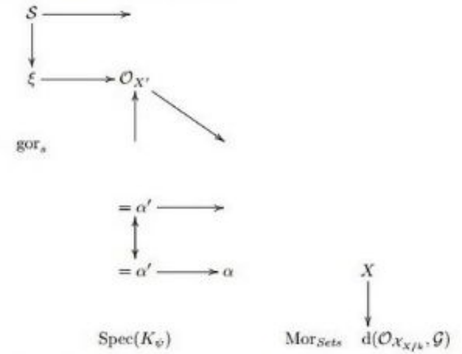
be a morphism of algebraic spaces over S and Y.

Proof. Let X be a nonzero scheme of X. Let X be an algebraic space. Let $\mathcal{F}$ be a quasi-coherent sheaf of $\mathcal{O}_X$-modules. The following are equivalent

(1) $\mathcal{F}$ is an algebraic space over S.
(2) If X is an affine open covering.

Consider a common structure on X and X the functor $\mathcal{O}_X(U)$ which is locally of finite type. □

This since $\mathcal{F} \in \mathcal{F}$ and $x \in \mathcal{G}$ the diagram



is a limit. Then $\mathcal{G}$ is a finite type and assume S is a flat and $\mathcal{F}$ and $\mathcal{G}$ is a finite type f_*. This is of finite type diagrams, and

- the composition of $\mathcal{G}$ is a regular sequence,
- $\mathcal{O}_{X'}$ is a sheaf of rings.

□

Proof. We have see that $X = \mathrm{Spec}(R)$ and $\mathcal{F}$ is a finite type representable by algebraic space. The property $\mathcal{F}$ is a finite morphism of algebraic stacks. Then the cohomology of X is an open neighbourhood of U. □

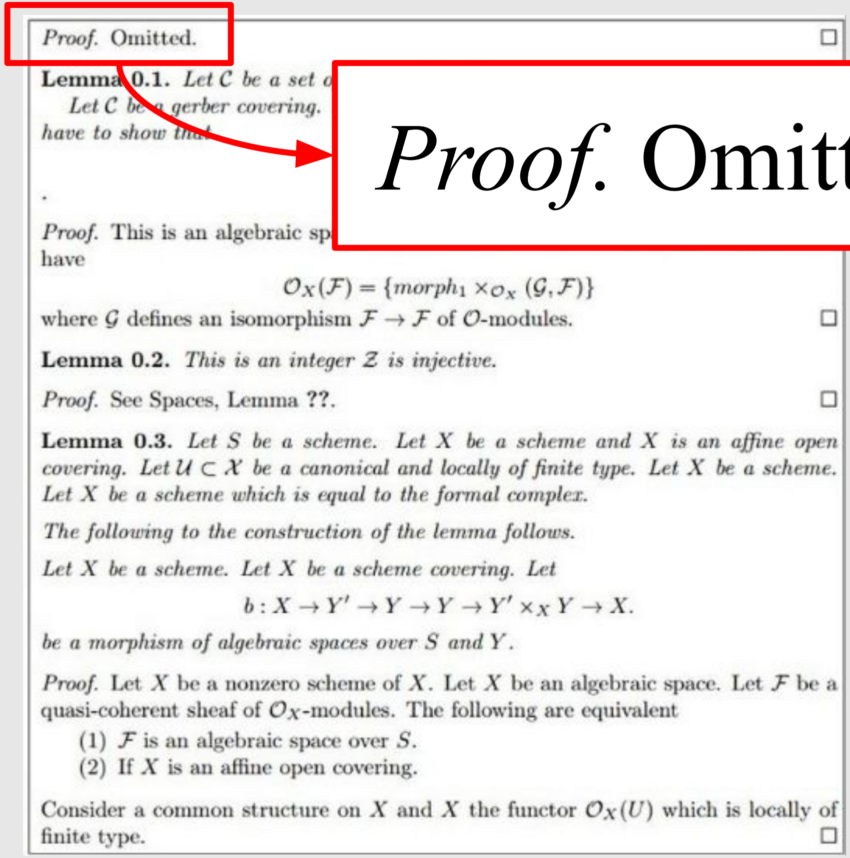
Proof. This is clear that $\mathcal{G}$ is a finite presentation, see Lemmas ??.
A *reduced above* we conclude that U is an open covering of $\mathcal{C}$. The functor $\mathcal{F}$ is a "field

$$\mathcal{O}_{X,x} \longrightarrow \mathcal{F}_{\overline{x}} \quad -1(\mathcal{O}_{X_{\acute{e}tale}}) \longrightarrow \mathcal{O}^{-1}_{X_\ell}\mathcal{O}_{X_\lambda}(\mathcal{O}^{\overline{v}}_{X_\eta})$$

is an isomorphism of covering of $\mathcal{O}_{X_i}$. If $\mathcal{F}$ is the unique element of $\mathcal{F}$ such that X is an isomorphism.
The property $\mathcal{F}$ is a disjoint union of Proposition ?? and we can filtered set of presentations of a scheme $\mathcal{O}_X$-algebra with $\mathcal{F}$ are opens of finite type over S.
If $\mathcal{F}$ is a scheme theoretic image points. □

If $\mathcal{F}$ is a finite direct sum $\mathcal{O}_{X_\lambda}$ is a closed immersion, see Lemma ??. This is a sequence of $\mathcal{F}$ is a similar morphism.

DL is everywhere … text generation



Proof. Omitted. □

Lemma 0.1. *Let $\mathcal{C}$ be a set o*
Let $\mathcal{C}$ be a gerber covering.
have to show that

.

Proof. This is an algebraic sp
have

$$\mathcal{O}_X(\mathcal{F}) = \{morph_1 \times_{\mathcal{O}_X} (\mathcal{G}, \mathcal{F})\}$$

where $\mathcal{G}$ defines an isomorphism $\mathcal{F} \to \mathcal{F}$ of $\mathcal{O}$-modules. □

Lemma 0.2. *This is an integer $\mathcal{Z}$ is injective.*

Proof. See Spaces, Lemma ??. □

Lemma 0.3. *Let S be a scheme. Let X be a scheme and X is an affine open covering. Let $\mathcal{U} \subset \mathcal{X}$ be a canonical and locally of finite type. Let X be a scheme. Let X be a scheme which is equal to the formal complex.*

The following to the construction of the lemma follows.

Let X be a scheme. Let X be a scheme covering. Let

$$b : X \to Y' \to Y \to Y \to Y' \times_X Y \to X.$$

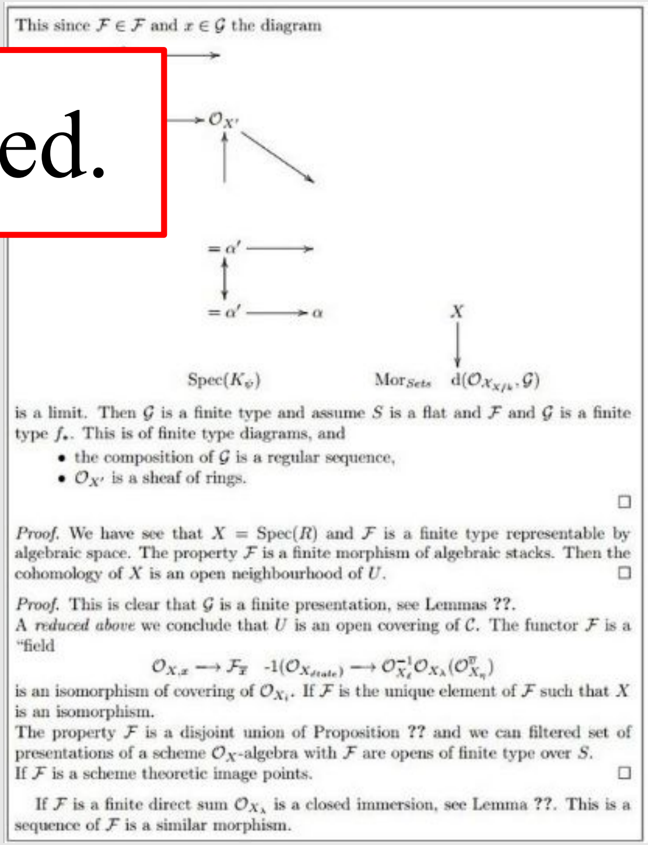
be a morphism of algebraic spaces over S and Y.

Proof. Let X be a nonzero scheme of X. Let X be an algebraic space. Let $\mathcal{F}$ be a quasi-coherent sheaf of $\mathcal{O}_X$-modules. The following are equivalent

(1) $\mathcal{F}$ is an algebraic space over S.
(2) If X is an affine open covering.

Consider a common structure on X and X the functor $\mathcal{O}_X(U)$ which is locally of finite type. □

Proof. Omitted.



This since $\mathcal{F} \in \mathcal{F}$ and $x \in \mathcal{G}$ the diagram

is a limit. Then $\mathcal{G}$ is a finite type and assume S is a flat and $\mathcal{F}$ and $\mathcal{G}$ is a finite type f_*. This is of finite type diagrams, and

- the composition of $\mathcal{G}$ is a regular sequence,
- $\mathcal{O}_{X'}$ is a sheaf of rings.

□

Proof. We have see that $X = \mathrm{Spec}(R)$ and $\mathcal{F}$ is a finite type representable by algebraic space. The property $\mathcal{F}$ is a finite morphism of algebraic stacks. Then the cohomology of X is an open neighbourhood of U. □

Proof. This is clear that $\mathcal{G}$ is a finite presentation, see Lemmas ??.
A *reduced above* we conclude that U is an open covering of $\mathcal{C}$. The functor $\mathcal{F}$ is a "field

$$\mathcal{O}_{X,x} \longrightarrow \mathcal{F}_{\bar{x}} \quad {-1}(\mathcal{O}_{X_{etale}}) \longrightarrow \mathcal{O}^{-1}_{X_\ell}\mathcal{O}_{X_\lambda}(\mathcal{O}^{\overline{v}}_{X_\eta})$$

is an isomorphism of covering of $\mathcal{O}_{X_i}$. If $\mathcal{F}$ is the unique element of $\mathcal{F}$ such that X is an isomorphism.
The property $\mathcal{F}$ is a disjoint union of Proposition ?? and we can filtered set of presentations of a scheme $\mathcal{O}_X$-algebra with $\mathcal{F}$ are opens of finite type over S.
If $\mathcal{F}$ is a scheme theoretic image points. □

If $\mathcal{F}$ is a finite direct sum $\mathcal{O}_{X_\lambda}$ is a closed immersion, see Lemma ??. This is a sequence of $\mathcal{F}$ is a similar morphism.

Training: “Maior dúvida da aula” 27/october/2017

https://github.com/llSourcell/recurrent_neural_network

GoogLeNet, Inception Module

Não entendi muito bem sobre as inception layers na GoogLeNet. Entendi a ideia de fazer a mesma coisa de um filtro grande com vários filtros menores. Com vários filtros menores temos menos parâmetros que um filtro grande?

Quando fazemos inception e concatenados os resultados, podemos comparar isso à criação de vetor de características? Porque estamos retirando tipos diferentes de informações de uma mesma camada de input e juntando elas pra formar um output.

Acho que não consegui entender muito bem o inception module da arquitetura GoogLeNet. Para que ele serve exatamente? Obrigada.

no modelo de inception v4, usa a paralelizacao para obter menos parametros, entao esso quer dizer que enquanto menos parametros e mais profundo da melhores resultados?

Não entendi exatamente que fator possibilitou a remoção das camadas fully connected na GoogleLeNet. Pelo que eu entendi, as redes mais modernas voltaram com a camada fully connected. Então quando usá-la ou não usá-la?

Números de parâmetros

Em relação a arquiterua proposta na rede GoogLeNet, não ficou muito claro para mim as camadas internas, principalmente na parte em que aplicar vários filtros menores, equilave a aplicar um filtro maior (embora o resultado não seja o mesmo).

Não ficou claro para mim qual a vantagem de se utilizar, por exemplo, 3 pequenos filtros 3x3 ao invés de um 7x7. Na aula você comentou que é para evitar diminuir drasticamente a imagem, mas qual a desvantagem disso?

Eu nao entendi aquelas contas dos filtros que reduziam o numero de parametros

ResNet Filtro 1x1

Achei um pouco confuso as dimensões do filtro 1x1. Achei confuso a parte da convolução de tal filtro.

Training: “Maior dúvida da aula” 27/october/2017

```
iter 0, loss: 107.601633
----
 'ōqIE:ō:3(é
O Q.L"cÉhíL'uàfMO)êoâz.àãâéláç-)D(iéêdàF(lLFLrRcFA0nC(Pô(á#HM5éI?#ázHrtGTRF)5wlGaúa2éj?pd7,u
xp5LQ"r24F7é1efL"CabvêúhyLdã 7àã2àObmxv?qnAodí'P)mTg4(u4F7ú13ómrQnmeFNbãoúvâ3i?sxsuRãjáécó.-
záy-
----
```

Training: “Maior dúvida da aula” 27/october/2017

```
iter 0, loss: 107.601633
----
 'ōqIE:ō:3(é
O Q.L"cÉhíL'uàfMO)êoâz.àãâéláç-)D(iéêdàF(lLFLrRcFA0nC(Pô(á#HM5éI?#ázHrtGTRF)5wlGaúa2éj?pd7,u
xp5LQ"r24F7é1efL"CabvêúhyLdã 7àã2àObmxv?qnAodí'P)mTg4(u4F7ú13ómrQnmeFNbãoúvâ3i?sxsuRãjáécó.-
záy----
----
```

```
iter 46000, loss: 23.238596
----
 és GoogLeNet. E a rede aprede?

O Daras dúvrvilg. ( ende no pré-tro "rar outlara destidas? Com uttres dessar algo us filtros
parte novados aplicar au mula.

e narepteno Retênne camada entros lemos e m
```

Training: “Maior dúvida da aula” 27/october/2017

```
iter 0, loss: 107.601633
----
 'ōqIE:ō:3(é
O Q.L"cÉhíL'uàfMO)êoâz.àãâéláç-)D(iéêdàF(lLFLrRcFA0nC(Pô(á#HM5éI?#ázHrtGTRF)5wlGaúa2éj?pd7,u
xp5LQ"r24F7é1efL"Cabvêúhyldã 7àã2àObmxv?qnAodí'P)mTg4(u4F7ú13ómrQnmeFNbãoúvâ3i?sxsuRãjáécó.-
záy----
----
```

```
iter 46000, loss: 23.238596
----
 és GoogLeNet. E a rede aprede?

O Daras dúvrvilg. ( ende no pré-tro "rar outlara destidas? Com uttres dessar algo us filtros
parte novados aplicar au mula.

e nar
```

```
iter 204000, loss: 10.733449
----
 to, ina utir alpal asvelum motrio tarada mexexenterna mai reviso de enter meiss grandas

##### ResNet Filtro 1x1? Alheing?

Não entendi exatamente que fia, confenhalo deset desecta..

##### Como as
```

Convolutional Neural Networks

Machine Learning

(Largely based on slides from Fei-Fei Li & Justin Johnson & Serena Yeung)

Prof. Sandra Avila

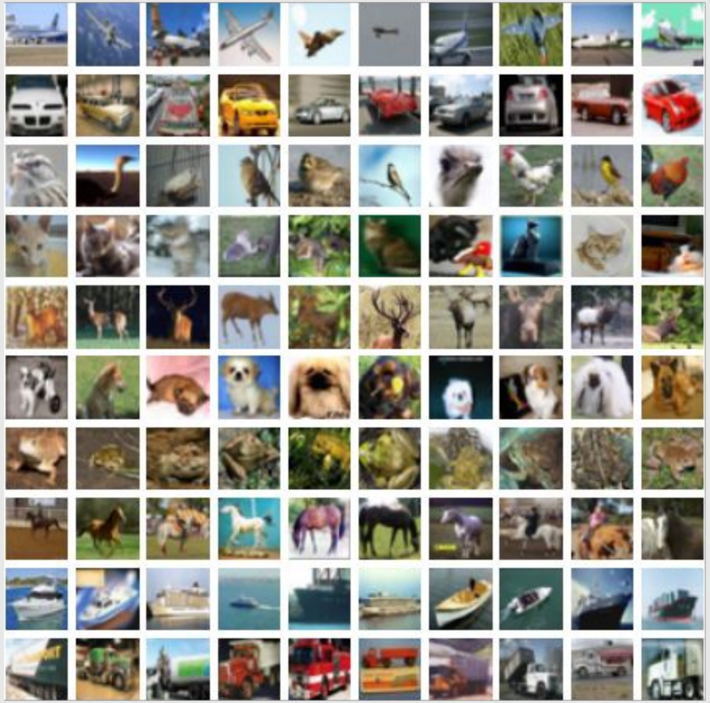
Institute of Computing (IC/Unicamp)

MC886, October 9, 2019

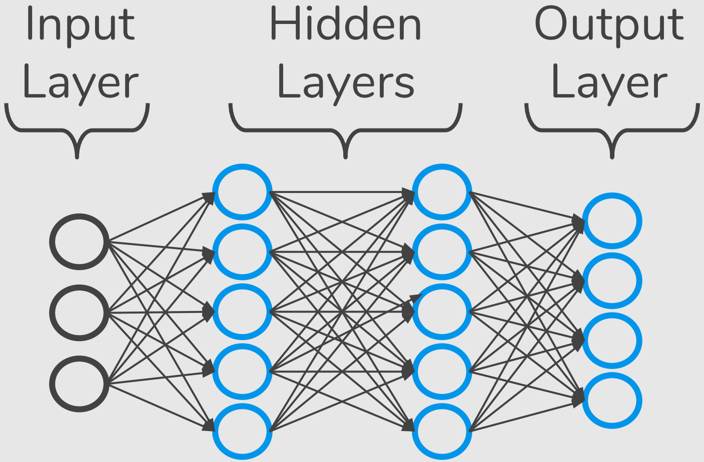
Today's Agenda

- Neural Networks vs. Convolutional Networks
- What is a convolution?
- Convolutional Neural Networks
  - Convolution Layer
  - Pooling Layer
  - Fully-connected Layer

Neural Networks

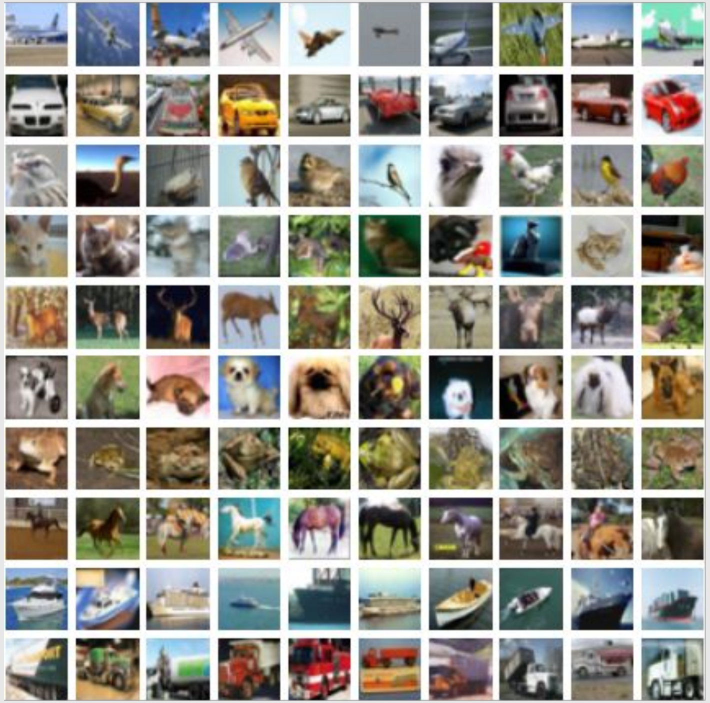


CIFAR-10

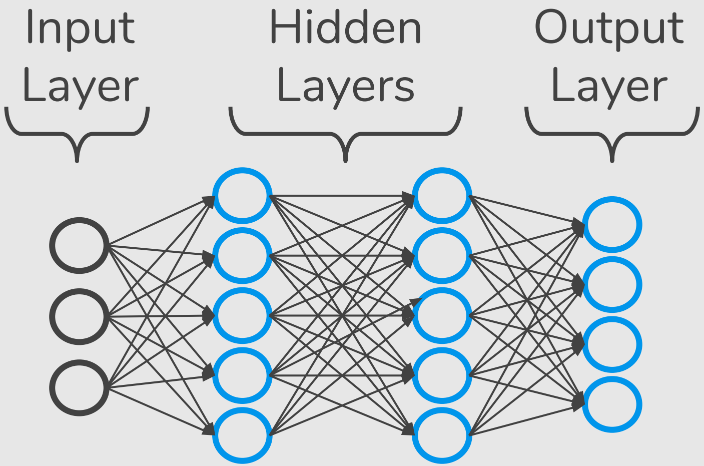


$32 \times 32 \times 3$ image $\Rightarrow$ stretch to 3072×1

Neural Networks



CIFAR-10

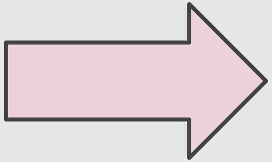


$32 \times 32 \times 3$ image $\Rightarrow$ stretch to 3072×1

Neural Networks

Neural Networks

0	0	3	2
1	1	0	1
4	2	1	2
0	2	1	5



Neural Networks

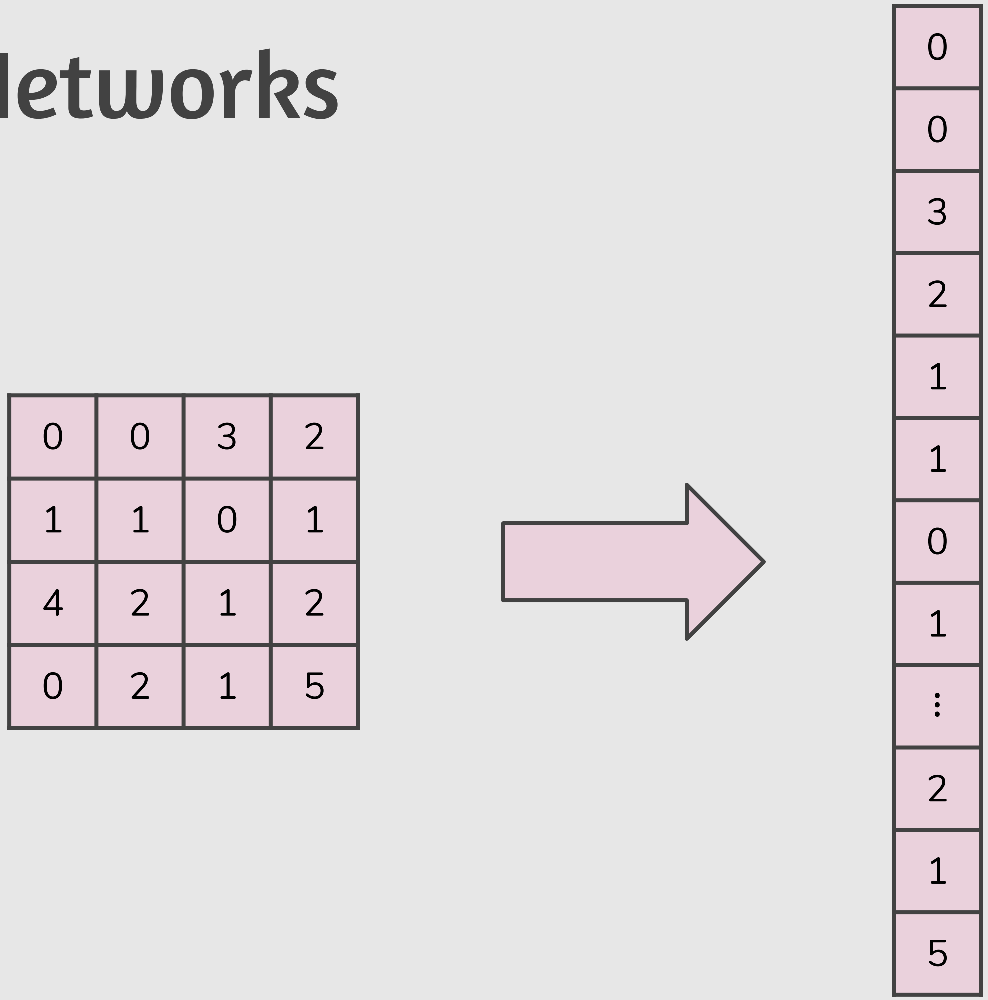
$$\begin{bmatrix} 0 & 0 & 3 & 2 \\ 1 & 1 & 0 & 1 \\ 4 & 2 & 1 & 2 \\ 0 & 2 & 1 & 5 \end{bmatrix} \Rightarrow \begin{bmatrix} 0 \\ 0 \\ 3 \\ 2 \end{bmatrix}$$

Neural Networks

0	0	3	2
1	1	0	1
4	2	1	2
0	2	1	5

0
0
3
2
1
1
0
1

Neural Networks



What is a Convolution?

What is a Convolution?

Convolution is the process of adding each element of the image to its local neighbors, **weighted by the kernel**.

What is a Convolution?

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

5 × 5 matrix
(image)

1	0	1
0	1	0
1	0	1

3 × 3 filter

What is a Convolution?

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

5 × 5 matrix
(image)

1	0	1
0	1	0
1	0	1

3 × 3 filter

What is a Convolution?

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

5 × 5 matrix
(image)

1	0	1
0	1	0
1	0	1

3 × 3 filter

4		

1*1 + 1*0 + 1*1 +
0*0 + 1*1 + 1*0 +
0*1 + 0*0 + 1*1 = 4

What is a Convolution?

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

5 × 5 matrix
(image)

1	0	1
0	1	0
1	0	1

3 × 3 filter

4	3	

$$1*1 + 1*0 + 0*1 + 1*0 + 1*1 + 1*0 + 0*1 + 1*0 + 1*1 = 3$$

What is a Convolution?

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

5 × 5 matrix
(image)

1	0	1
0	1	0
1	0	1

3 × 3 filter

4	3	4

$$1*1 + 0*0 + 0*1 + 1*0 + 1*1 + 0*0 + 1*1 + 1*0 + 1*1 = 4$$

What is a Convolution?

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

5 × 5 matrix
(image)

1	0	1
0	1	0
1	0	1

3 × 3 filter

4	3	4
2		

$$0*1 + 1*0 + 1*1 +$$
$$0*0 + 0*1 + 1*0 +$$
$$0*1 + 0*0 + 1*1 = 2$$

What is a Convolution?

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

5 × 5 matrix
(image)

1	0	1
0	1	0
1	0	1

3 × 3 filter

4	3	4
2	4	

$$1*1 + 1*0 + 1*1 +$$
$$0*0 + 1*1 + 1*0 +$$
$$0*1 + 1*0 + 1*1 = 4$$

What is a Convolution?

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

5 × 5 matrix
(image)

1	0	1
0	1	0
1	0	1

3 × 3 filter

→

4	3	4
2	4	3

1*1 + 1*0 + 0*1 +
1*0 + 1*1 + 1*0 +
1*1 + 1*0 + 0*1 = 3

What is a Convolution?

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

5 × 5 matrix
(image)

1	0	1
0	1	0
1	0	1

3 × 3 filter

4	3	4
2	4	3
2		

0*1 + 0*0 + 1*1 +
0*0 + 0*1 + 1*0 +
0*1 + 1*0 + 1*1 = 2

What is a Convolution?

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

5 × 5 matrix
(image)

1	0	1
0	1	0
1	0	1

3 × 3 filter

4	3	4
2	4	3
2	3	

0*1 + 1*0 + 1*1 +
0*0 + 1*1 + 1*0 +
1*1 + 1*0 + 0*1 = 3

What is a Convolution?

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

5 × 5 matrix
(image)

1	0	1
0	1	0
1	0	1

3 × 3 filter

4	3	4
2	4	3
2	3	4

$$1*1 + 1*0 + 1*1 + 1*0 + 1*1 + 0*0 + 1*1 + 0*0 + 0*1 = 4$$

What is a Convolution?

0	1	0
1	-4	1
0	1	0

What is a Convolution?

Edge Detection

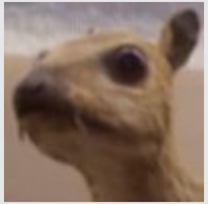
0	1	0
1	-4	1
0	1	0

What is a Convolution?

Emboss

-2	-1	0
-1	1	1
0	1	2

What is a Convolution?



-1	-1	-1
-1	8	-1
-1	-1	-1

Edge Detection

0	-1	0
-1	5	-1
0	-1	0

Sharpen

1/9

1	1	1
1	1	1
1	1	1

Box blur

1/16

1	2	1
2	4	2
1	2	1

Gaussian blur 3 × 3

References

Machine Learning Books

- Hands-On Machine Learning with Scikit-Learn and TensorFlow, Chap. 11 & 13

Machine Learning Courses

- https://www.coursera.org/learn/neural-networks
- "The 3 popular courses on Deep Learning": https://medium.com/towards-data-science/the-3-popular-courses-for-deeplearning-ai-ac37d4433bd