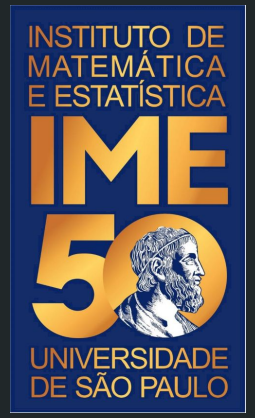
Sejam bem-vindas, sejam bem vindos!

Visão Computacional e Aprendizado Profundo

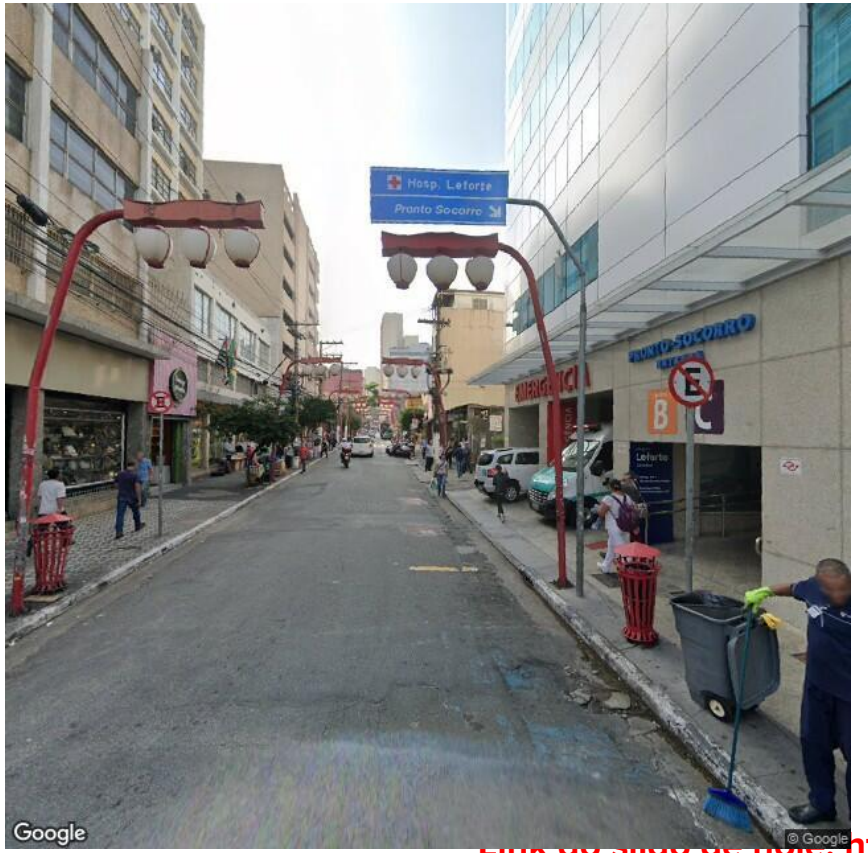


Oito anos da AlexNet

R. Hirata Jr.
IME - USP

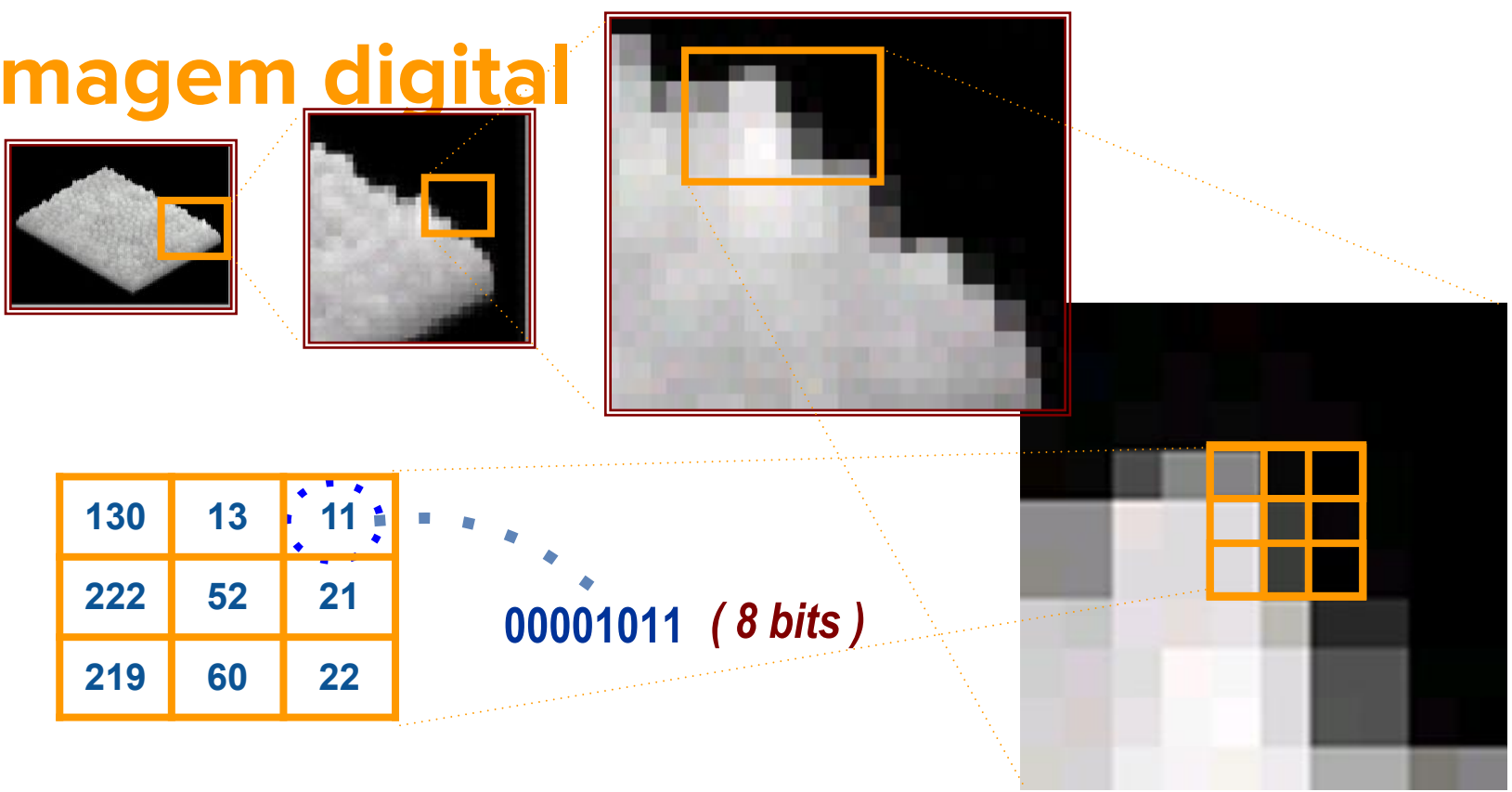
Link do slido de hoje: https://app.sli.do/event/bldcpzub

Em 2003 um estudante me disse: “Eu queria...”



Link do slido de hoje: https://app.sli.do/event/bldcpzub

Imagem digital



Pixel - *Picture Element*

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Propriedades da imagem

- Atributos físicos
  - Resolução espacial
    - Quantos pixels tem a imagem ?
  - Faixa dinâmica – ou resolução de amplitude
    - Quantos bits/pixels tem a imagem ?
  - Tamanho
    - Quanta memória a imagem ocupa ?
      - Tamanho = (Resolução) x (Faixa Dinâmica)
- Atributos perceptuais
  - Qualidades percebidas pela visão

Link do slido de hoje: https://app.sli.do/event/bldcpzub

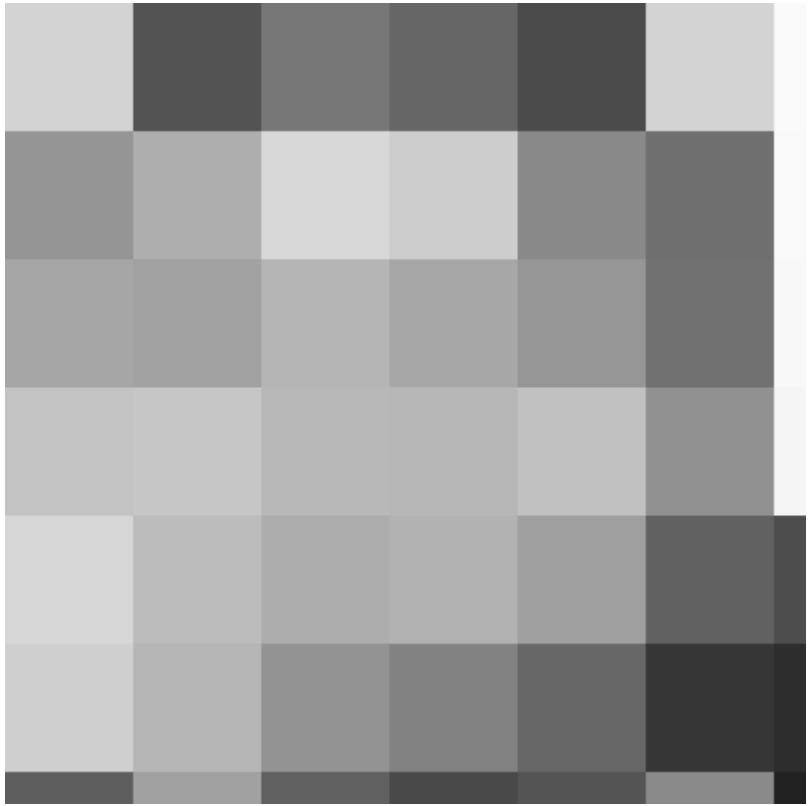
Níveis de Intensidades

Original

1 bit / pixel

Link do slido de hoje: https://app.sli.do/event/bldcpzub

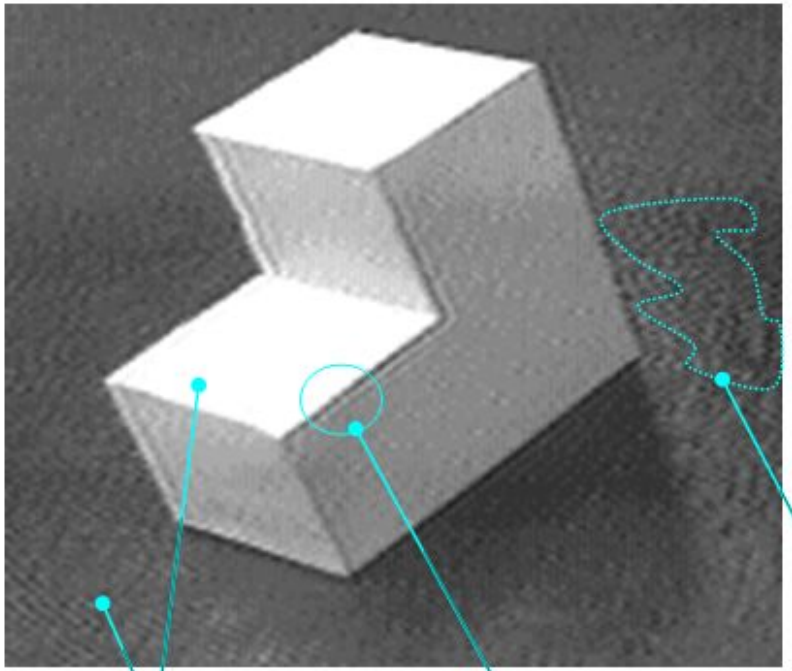
Resolução Espacial

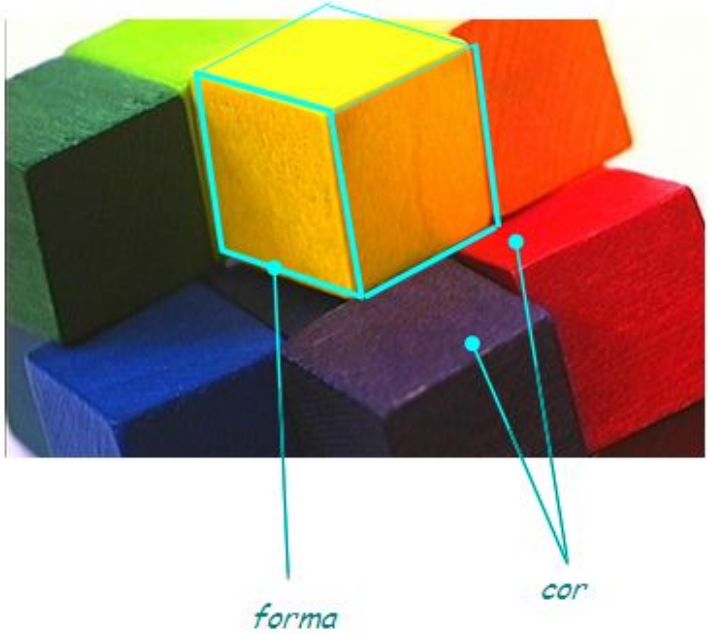


1 :64

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Atributos perceptuais da visão





Link do slido de hoje: https://app.sli.do/event/bldcpzub

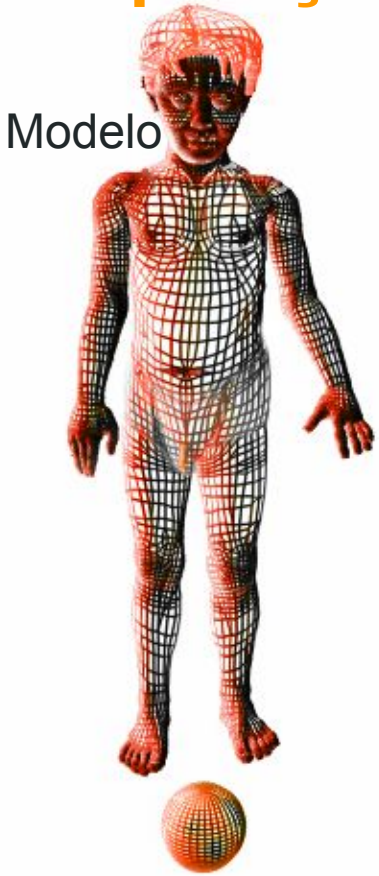
Processos

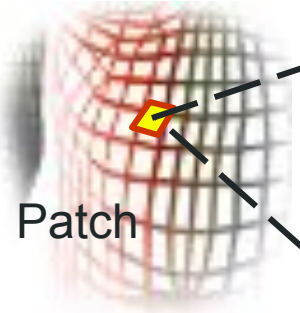
- *Processamento de Imagens*

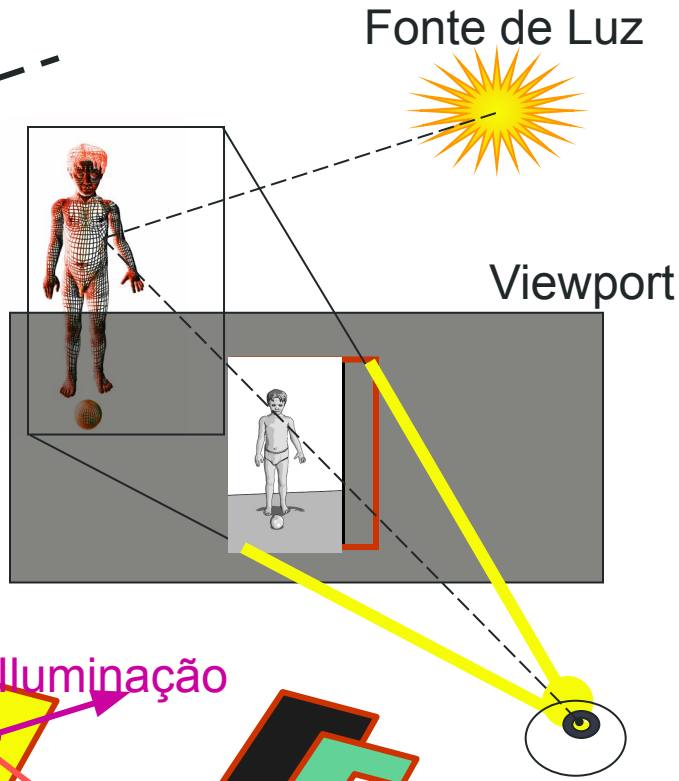
Processos

- *Computação Gráfica*

Computação gráfica



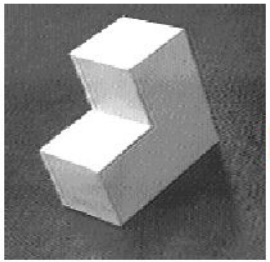




Link do slido de hoje: https://app.sli.do/event/bldcpzub

Processos

- ***Visão Computacional***

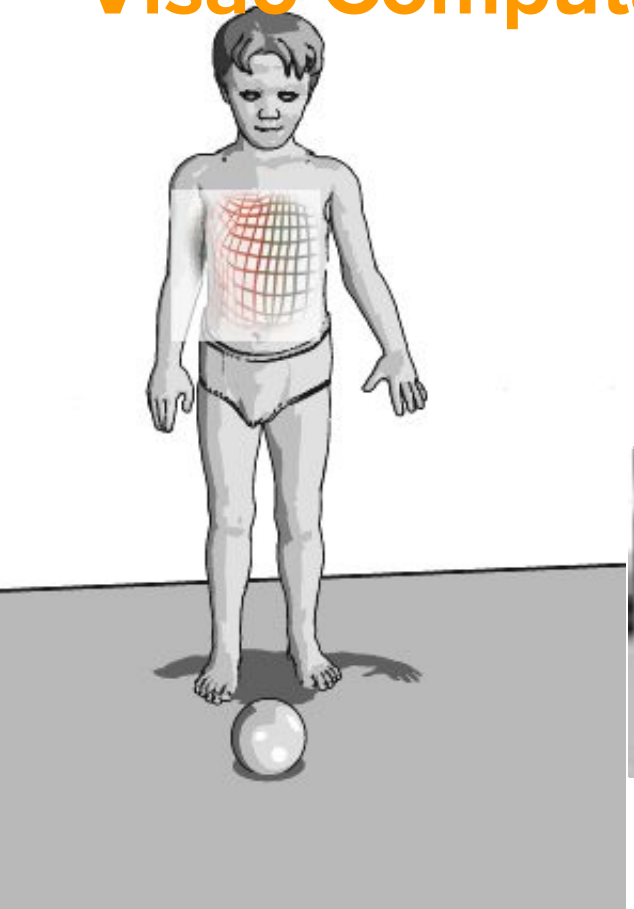


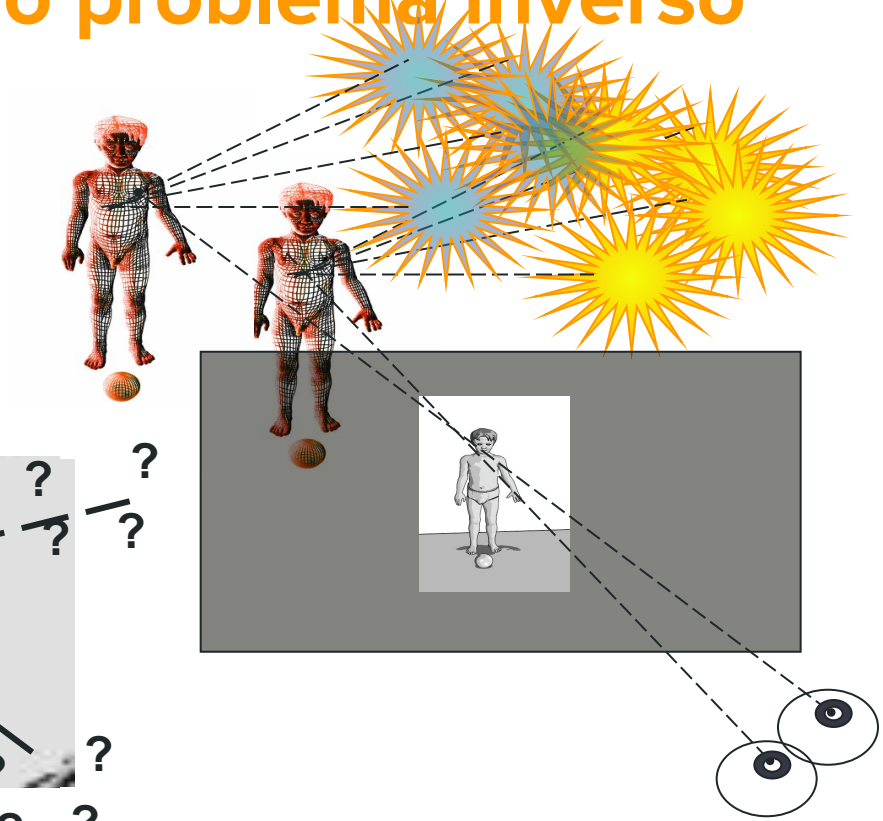
bloco
poliédrico
8 faces
pos. observador
pos. iluminação
fundo
etc

IMAGEM

DESCRIÇÃO
FORMAL

Visão Computacional e o problema inverso





Link do slido de hoje: https://app.sli.do/event/bldcpzub

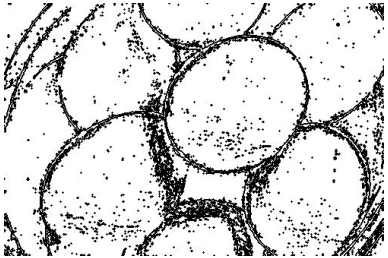
Visão Computacional

- Aplica-se a diversos tipos de problemas
  - Inspeção, identificação, monitoração, rastreamento, mensuração, etc
- Pesquisam-se representações e modelos adequados para cada problema
  - Problemas de aplicação tecnológica
  - Problemas de modelagem comportamental e/ou fisiológica

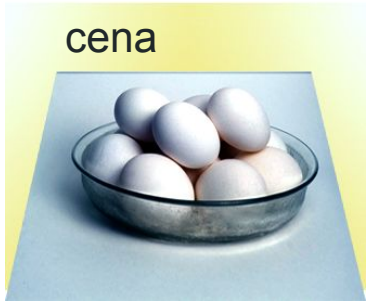
Link do slido de hoje: https://app.sli.do/event/bldcpzub

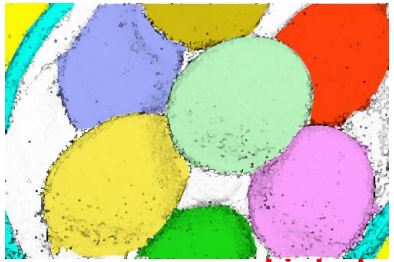
Visão Computacional

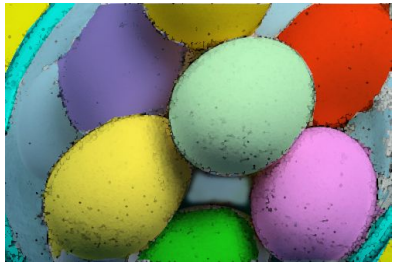
câmera



- Ovos
- Tigela
- Borda
- Fundo





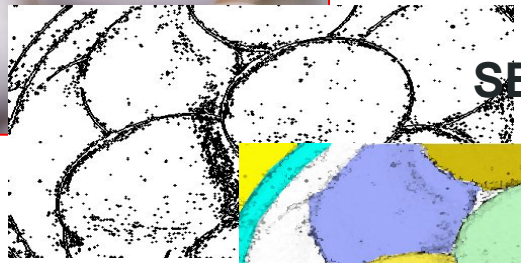


Link do slido de hoje: https://app.sli.do/event/bldcpzub

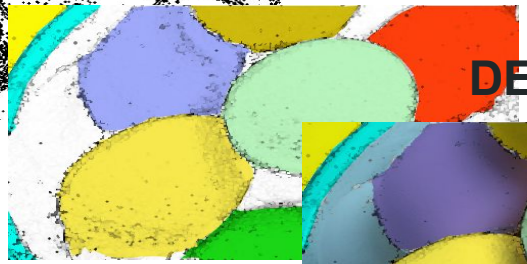
Visão Computacional

IMAGEAMENTO

PRÉ-PROCESSAMENTO



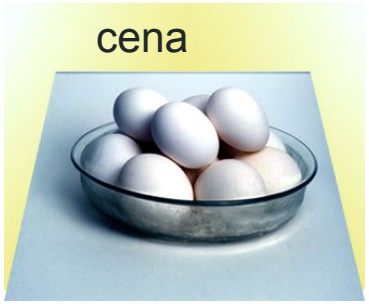
SEGMENTAÇÃO



DESCRIÇÃO

RECONHECIMENTO

câmera



- Ovos
- Tigela
- Borda
- Fundo

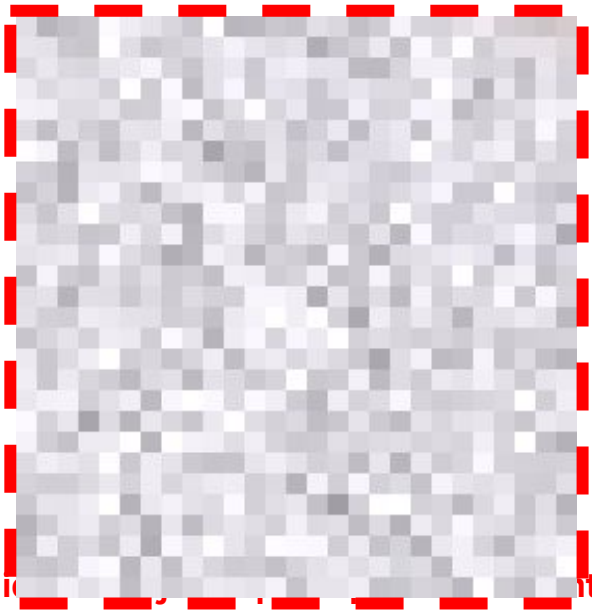
INTERPRETAÇÃO

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Pré-processamento

Imageamento

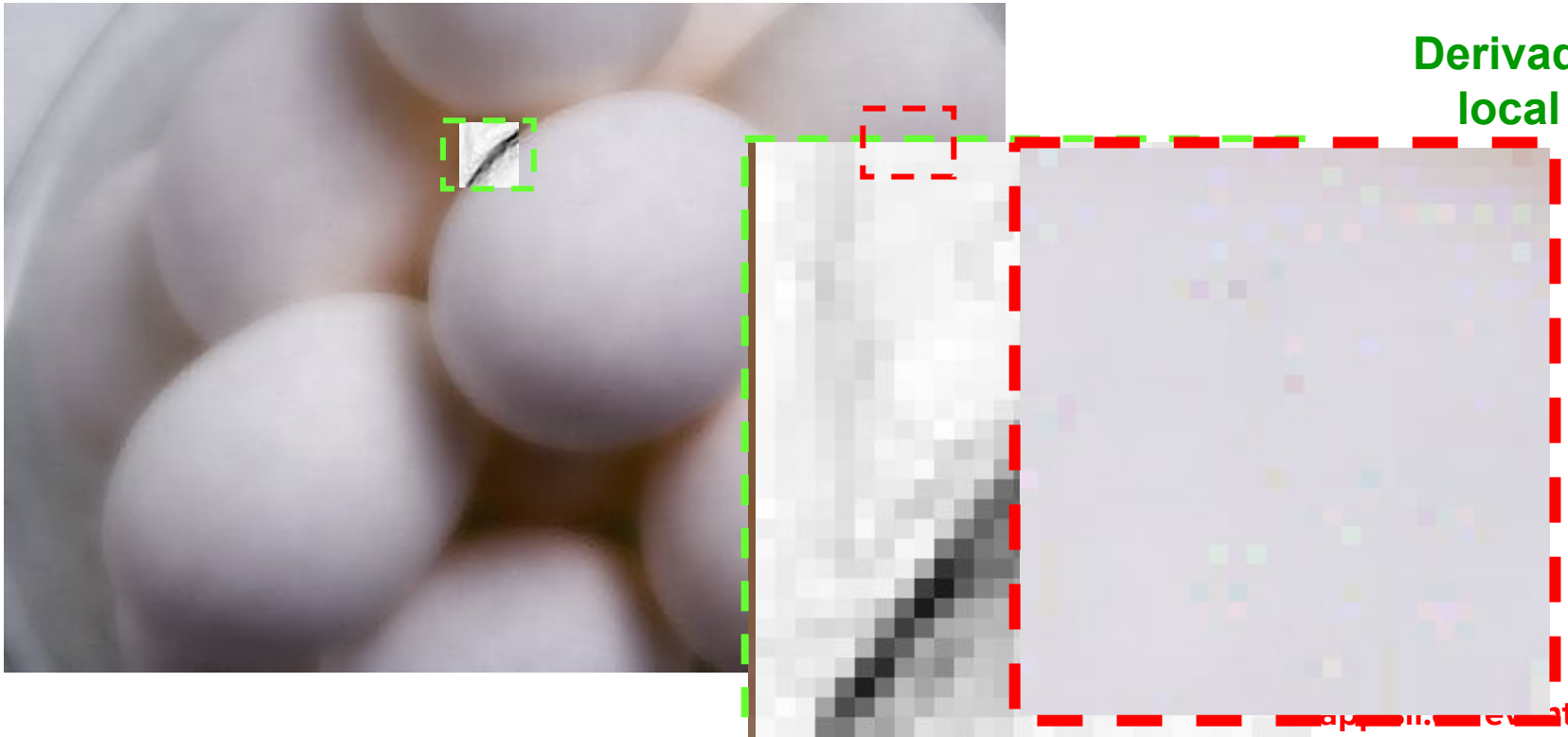
Média local



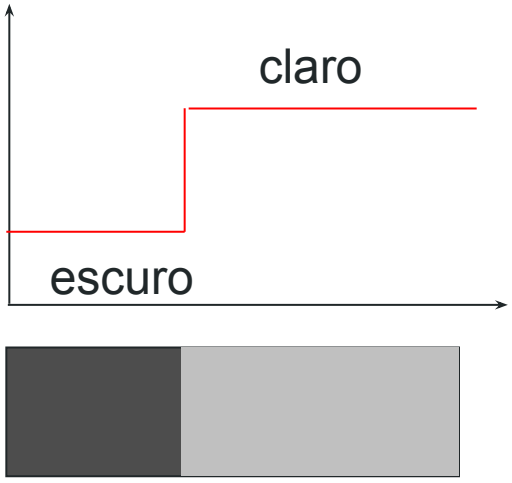
Link do sli[illegible]nt/bldcpzub

Pré-processamento

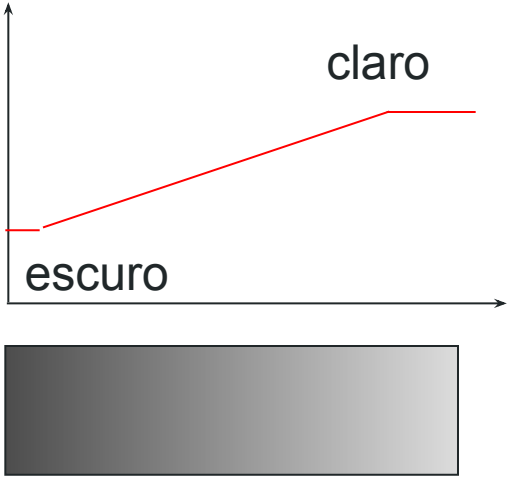
Segmentação



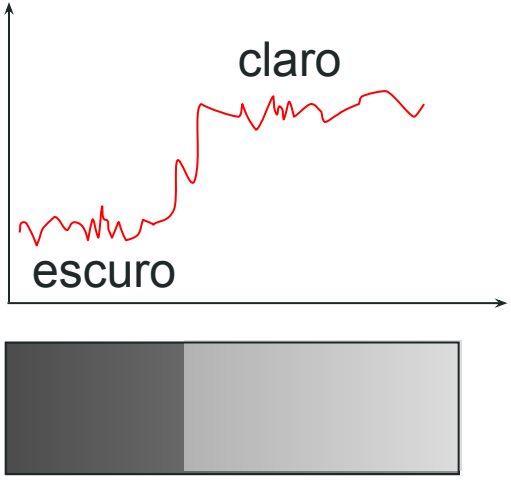
Princípio da detecção de contorno pelo gradiente local



Alto contraste
e sem ruído



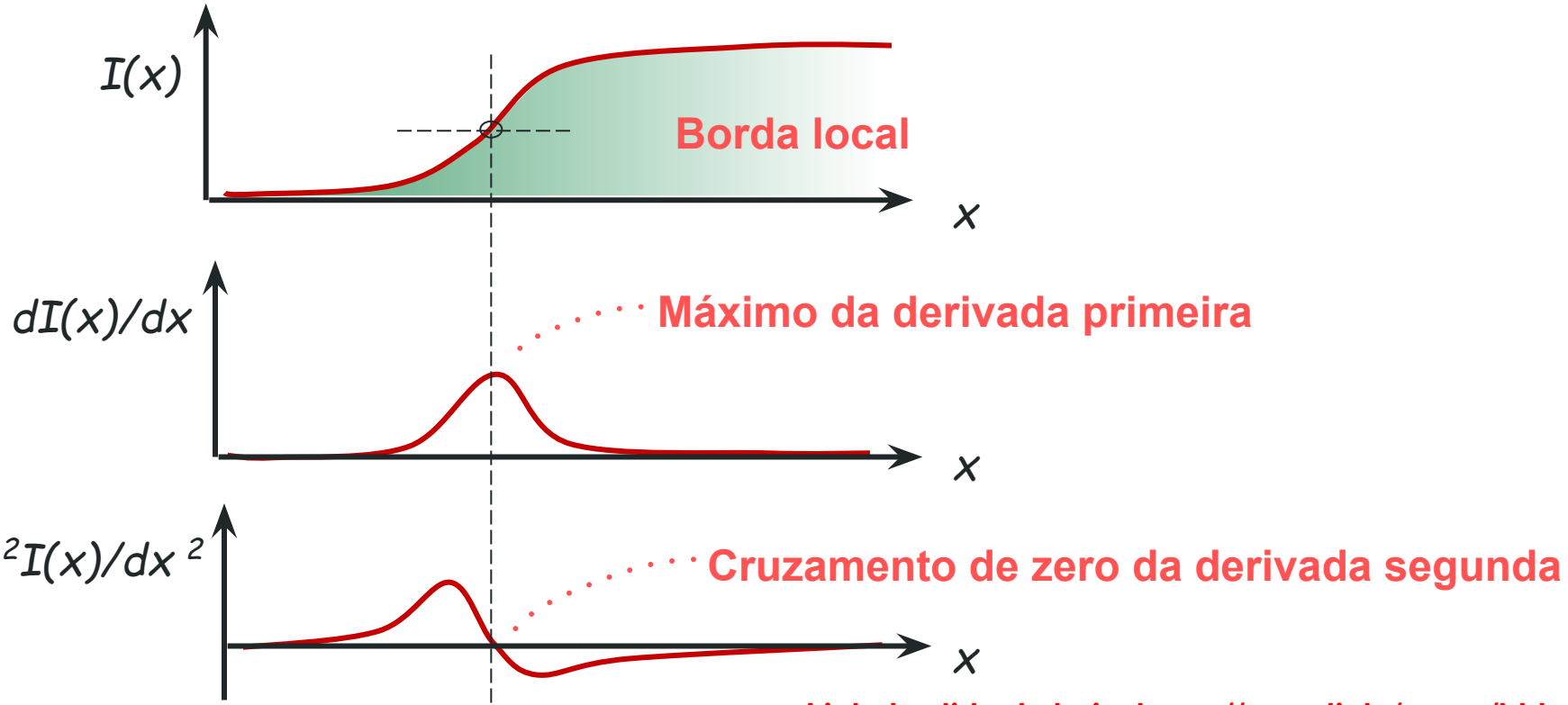
Baixo contraste
e sem ruído



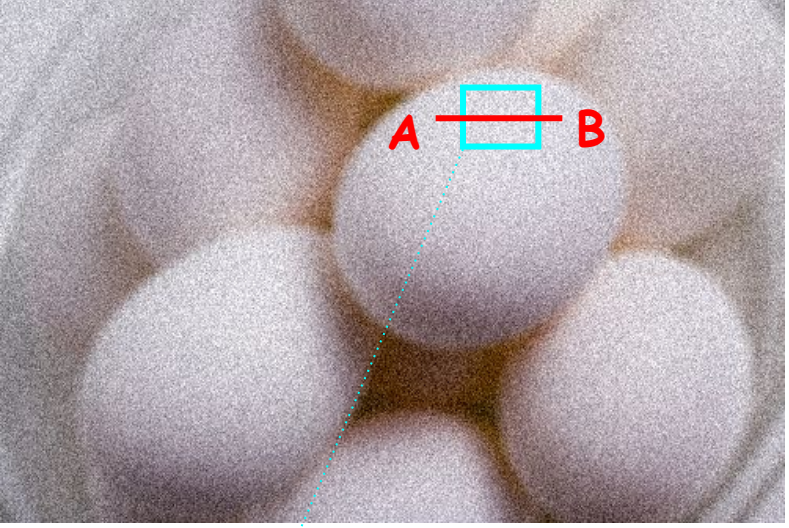
Bom contraste
e com ruído

Link do slido de hoje: https://app.sli.do/event/bldcpzub

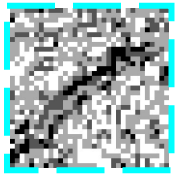
Princípio da detecção de contorno pelo gradiente local

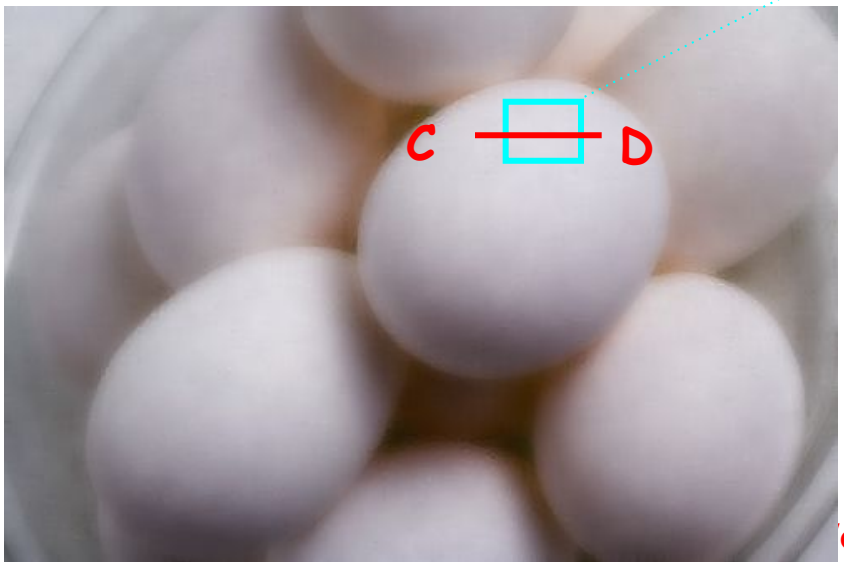


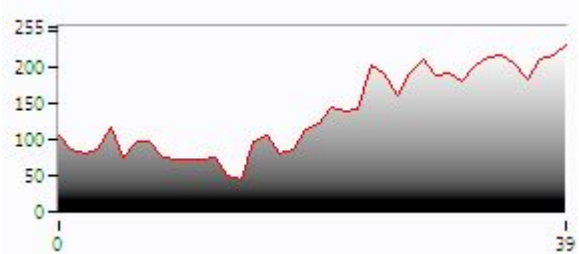
Link do slido de hoje: https://app.sli.do/event/bldcpzub



Perfil de intensidades ao longo de CD





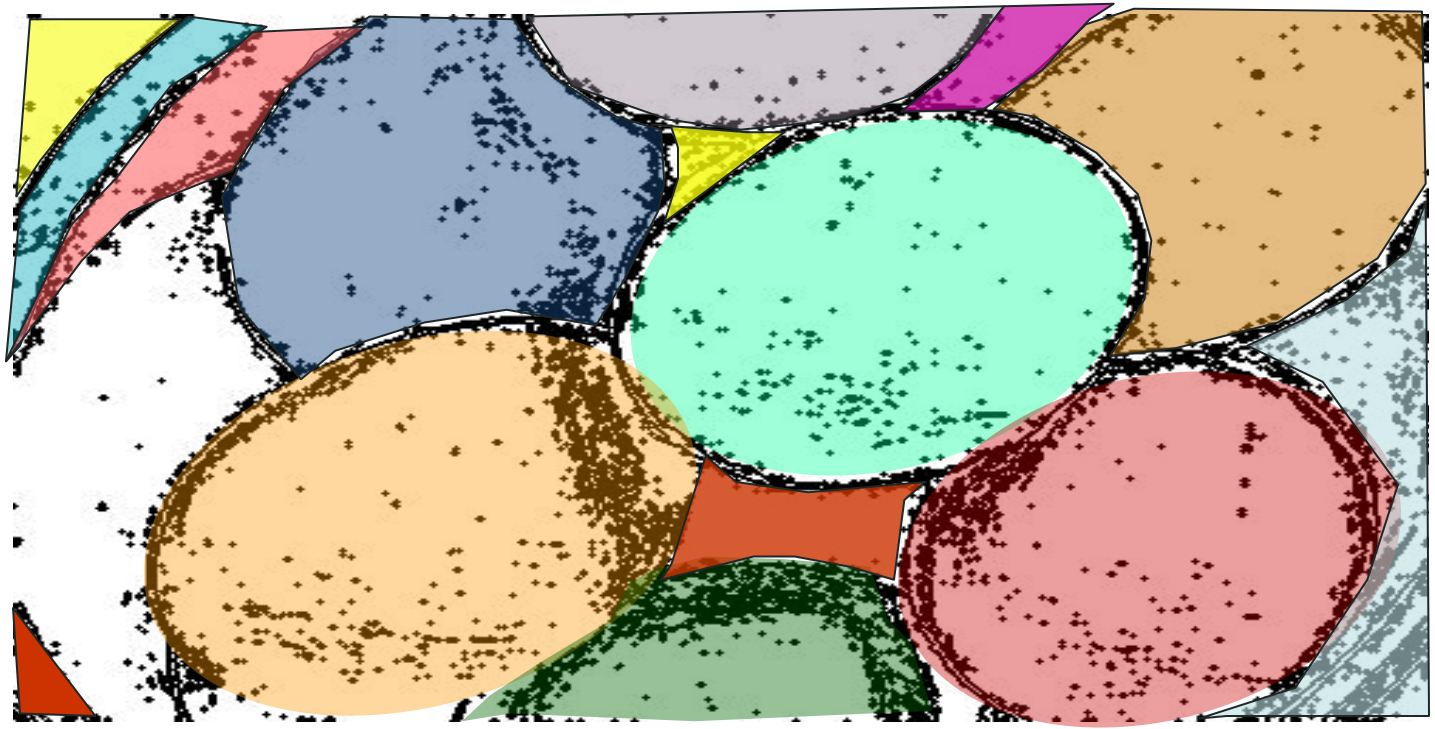


Perfil de intensidades ao longo de AB

event/bldcpzub

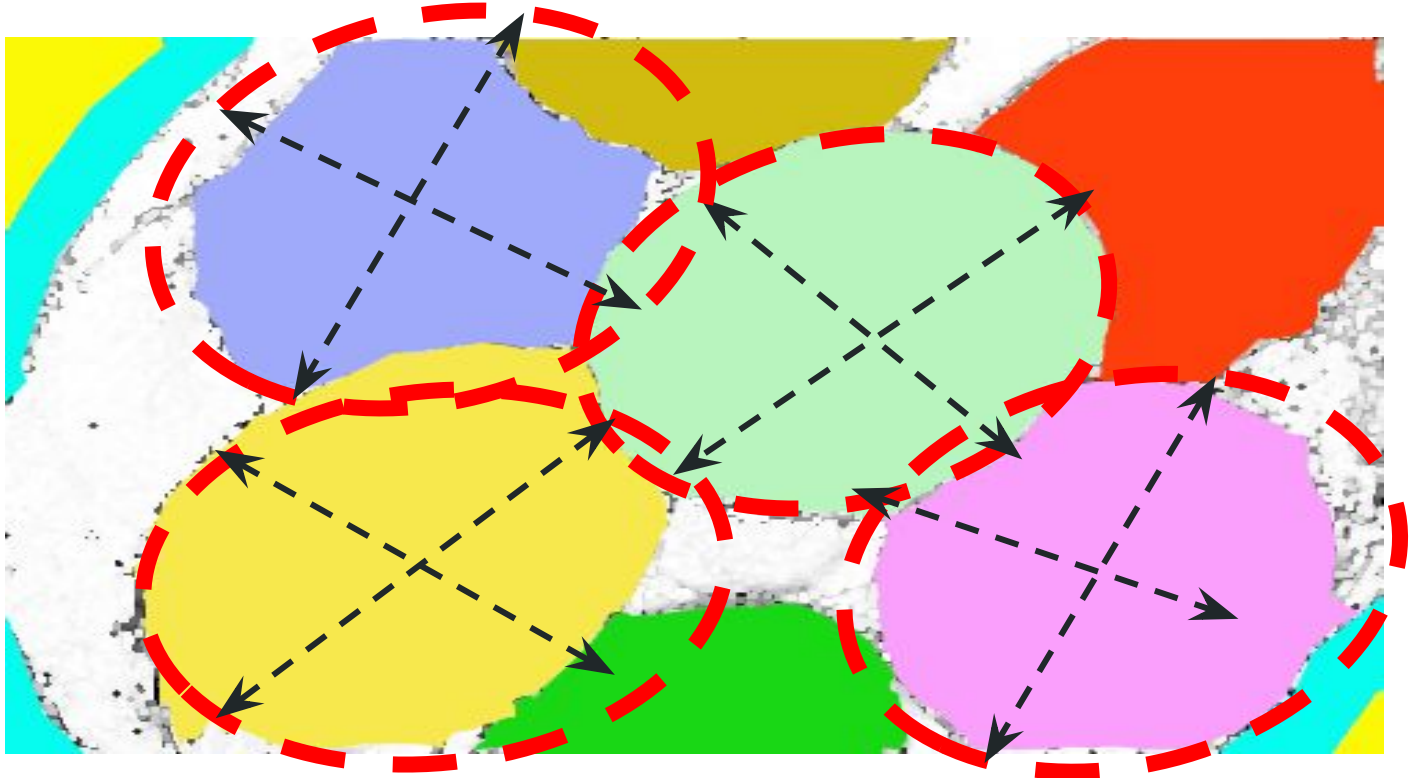
Segmentação

Descrição



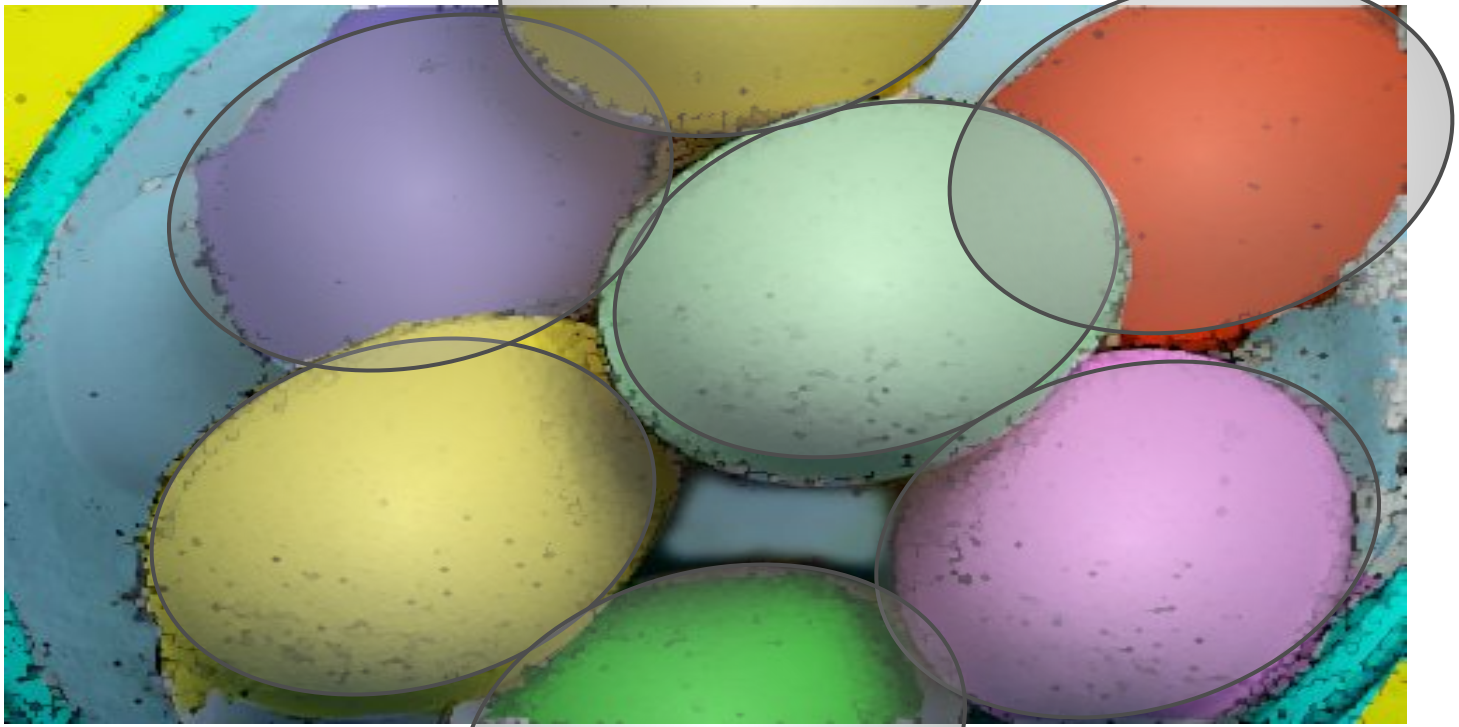
Link do slido de hoje: https://app.sli.do/event/bldcpzub

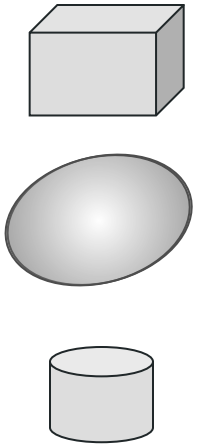
Descrição



Link do slido de hoje: https://app.sli.do/event/bldcpzub

Reconhecimento





Link do slido de hoje: https://app.sli.do/event/bldcpzub

Interpretação

- Ovos
- Tigela
- Borda
- Fundo

conhecimento

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Tarefas de Visão Computacional
e
Processamento de Imagens

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Algumas tarefas em Processamento de Imagens

- Análise de imagens no domínio das frequências
- Análise multirresolução de imagens
- Análise de formas binárias
- Binarização de imagens
- Detecção de bordas, linhas, círculos etc
- Detecção de buracos, cantos, polígonos etc
- Diminuição/Aumento de resolução
- Estimação de fluxo óptico
- Filtragem de ruído
- Segmentação de imagens
- Transformações geométricas (escala, espelhamento, rotação, translação etc)
- etc

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Algumas tarefas em Visão Computacional

- Descrição de cenas
- Deteção de objetos (animais, aviões, bicicletas, carros, drones, faces, pessoas etc)
- Identificação/reconhecimento de caracteres impressos e manuscritos
- Identificação/reconhecimento de impressão digital
- Identificação/reconhecimento de objetos
- Localização de imagens
- Rastreamento de objetos em movimento
- Restauração de imagens (inpainting, aumento de resolução)
- Segmentação de exemplos
- Vigilância
- etc

Link do slido de hoje: https://app.sli.do/event/bldcpzub

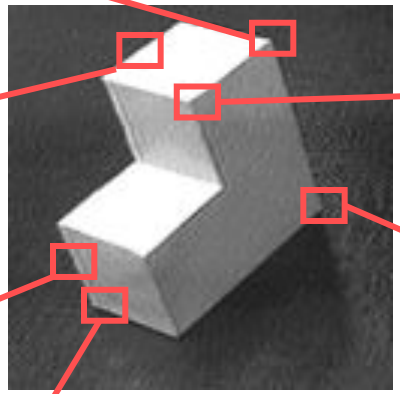
Atributos Locais

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Atributos locais (*Local features*)

Canto (corner)

Borda (edge)



Junção (junction)

Borda (edge)

Canto (corner)

Junção (junction)

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Atributos locais (*Local features*)

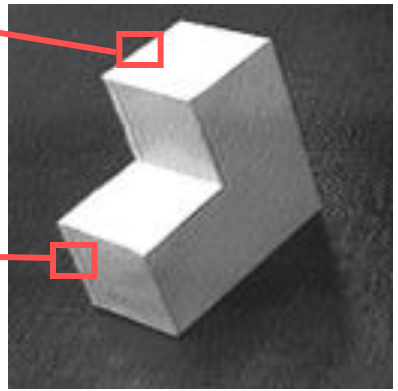
- São detalhes definidos para vizinhanças “pequenas” em torno de um pixel
  - O conceito de pequeno é relativo à resolução da imagem (largura x altura)
  - Dependendo da complexidade do atributo requer-se vizinhanças maiores
  - Define-se para cada tipo de atributo a menor vizinhança requerida para descrever o atributo

Link do slido de hoje: https://app.sli.do/event/bldcpzub

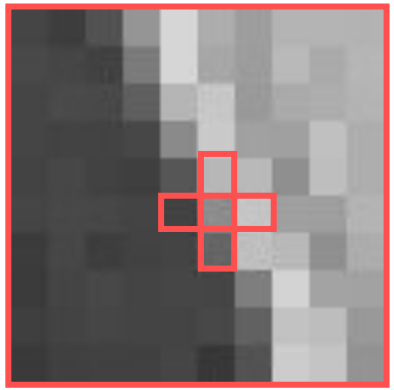
Atributos locais (*Local features*)

Borda (edge)

Borda (edge)



Bordas podem ser representadas com vizinhanças-4 ou 8 de 1ª ordem



Diferentes orientações

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Convolução

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

- É o nome generalizado para uma multiplicação de um vetor multidimensional por outro vetor multidimensional normalmente conhecido por máscara de tamanho menor, ou igual, ao primeiro vetor.
- Os pesos são os elementos que constituem a máscara de convolução.
- A filtragem convolucional é uma operação linear.
- A primeira palavra do nome “Convolution Neural Network” deve-se ao fato que a máscara “transladada” sobre a imagem
- No contexto de processamento de imagens, a convolução é na verdade uma correlação, mas isso não é realmente relevante

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

6	16	15	12	11	11	11	10	10	12	10
8	10	9	10	12	13	13	11	10	9	9
8	8	9	12	10	9	8	10	11	8	9
61	16	12	10	6	8	9	10	10	6	8
175	129	78	52	22	15	13	7	9	12	8
149	172	151	154	128	66	20	10	8	12	9
40	65	114	139	127	61	14	12	14	11	10
6	5	18	27	31	19	12	13	10	9	11
8	9	7	7	7	6	7	8	9	12	12
11	11	8	10	9	8	10	11	10	6	12
13	13	9	9	9	12	10	7	9	10	10

Imagem original - f

1/9	**1/9**	**1/9**
1/9	**1/9**	**1/9**
1/9	**1/9**	**1/9**

Máscara - g

0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0

Imagem convoluída - f * g

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

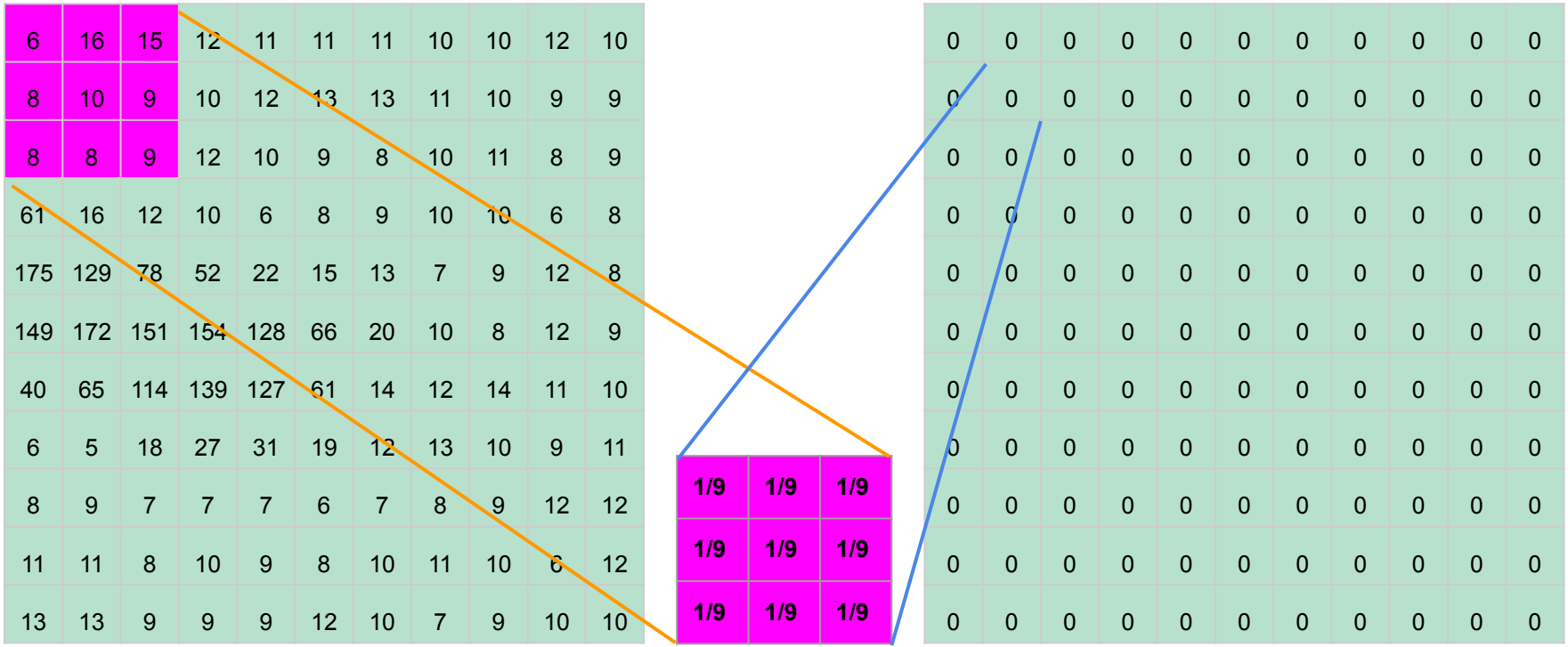


Imagem original - f **Máscara - g** **Imagem convoluída - f * g**

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

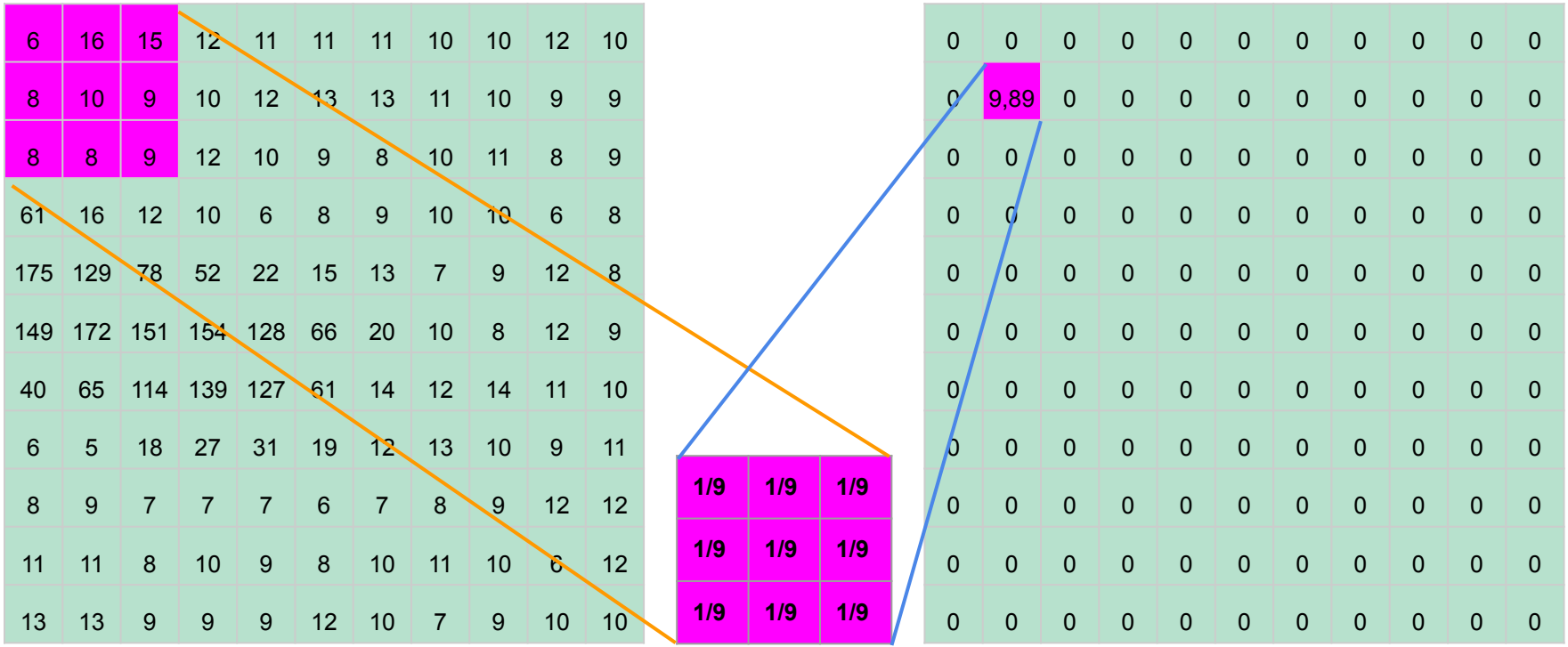


Imagem original - f **Máscara - 9** **Imagem convoluída - f * 9**

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

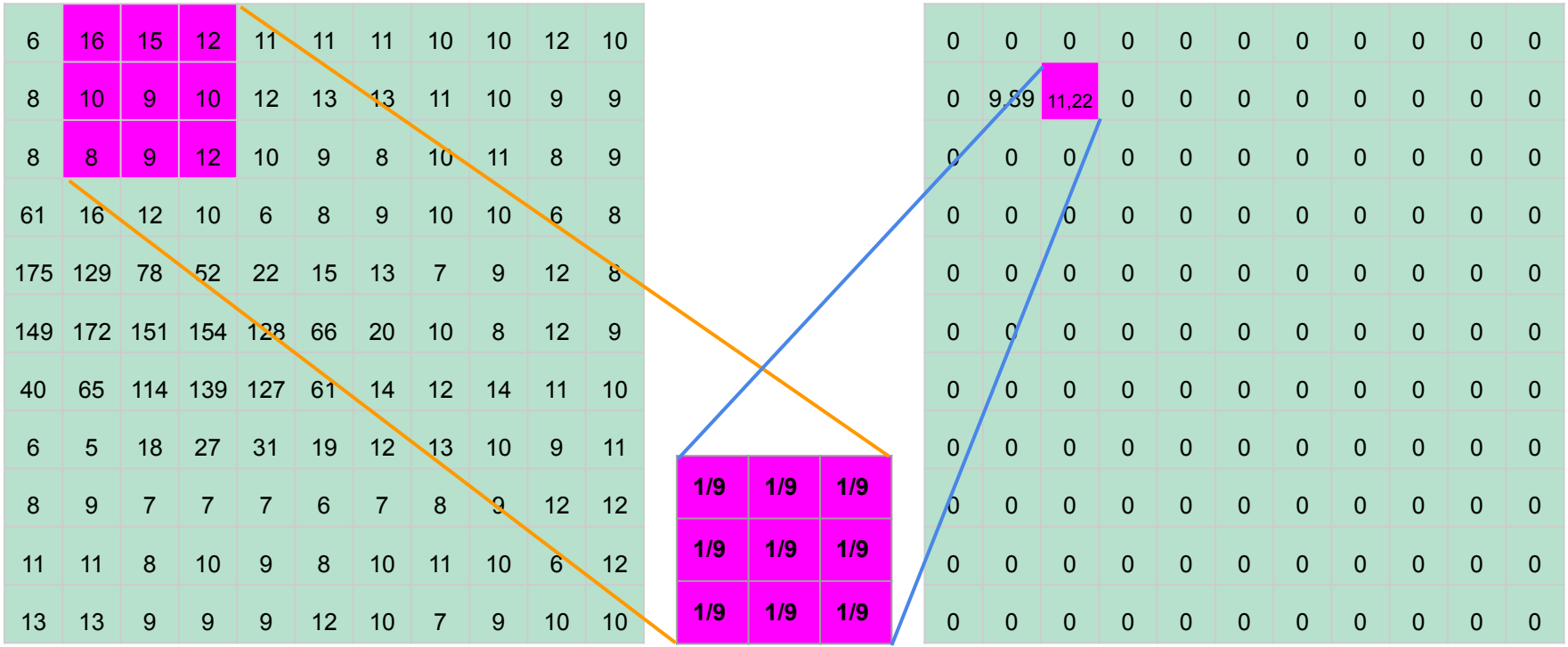


Imagem original - f **Máscara - g** **Imagem convoluída - f * g**

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

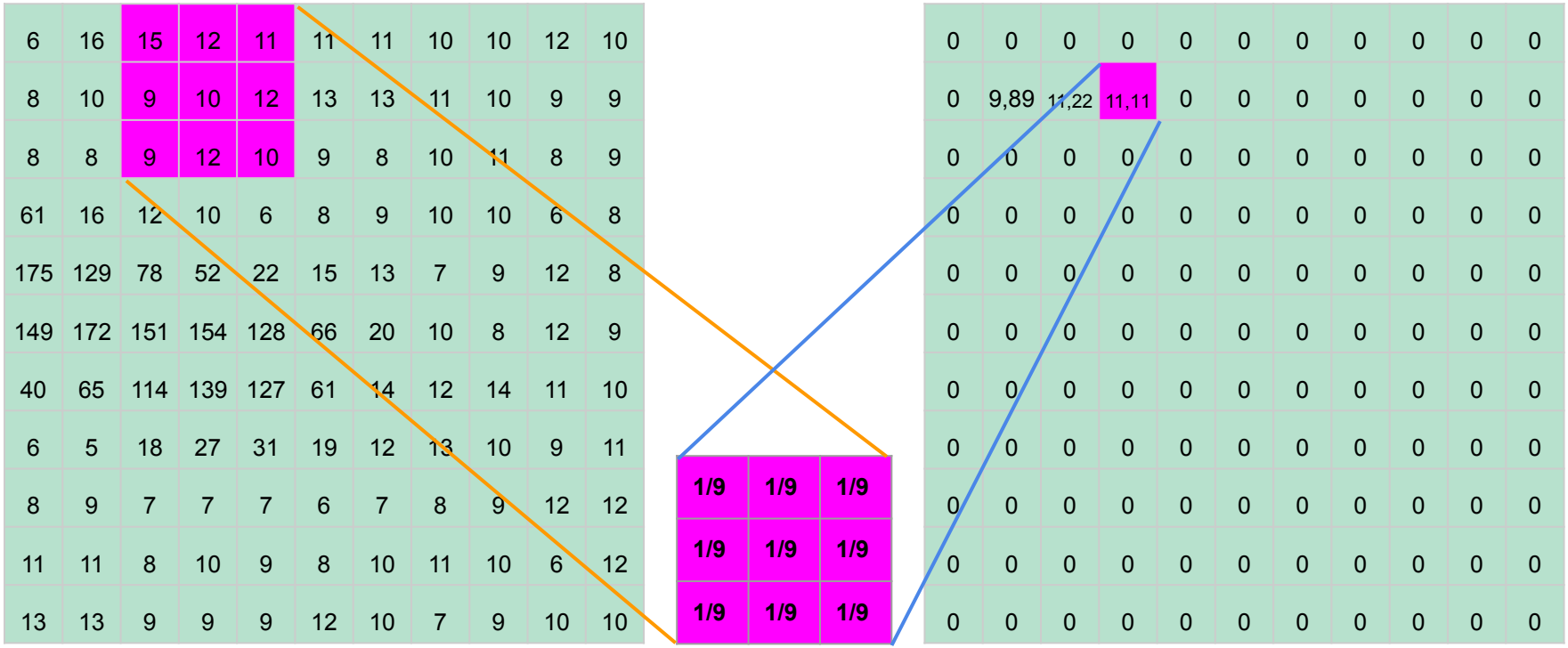


Imagem original - f **Máscara - g** **Imagem convoluída - f * g**

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

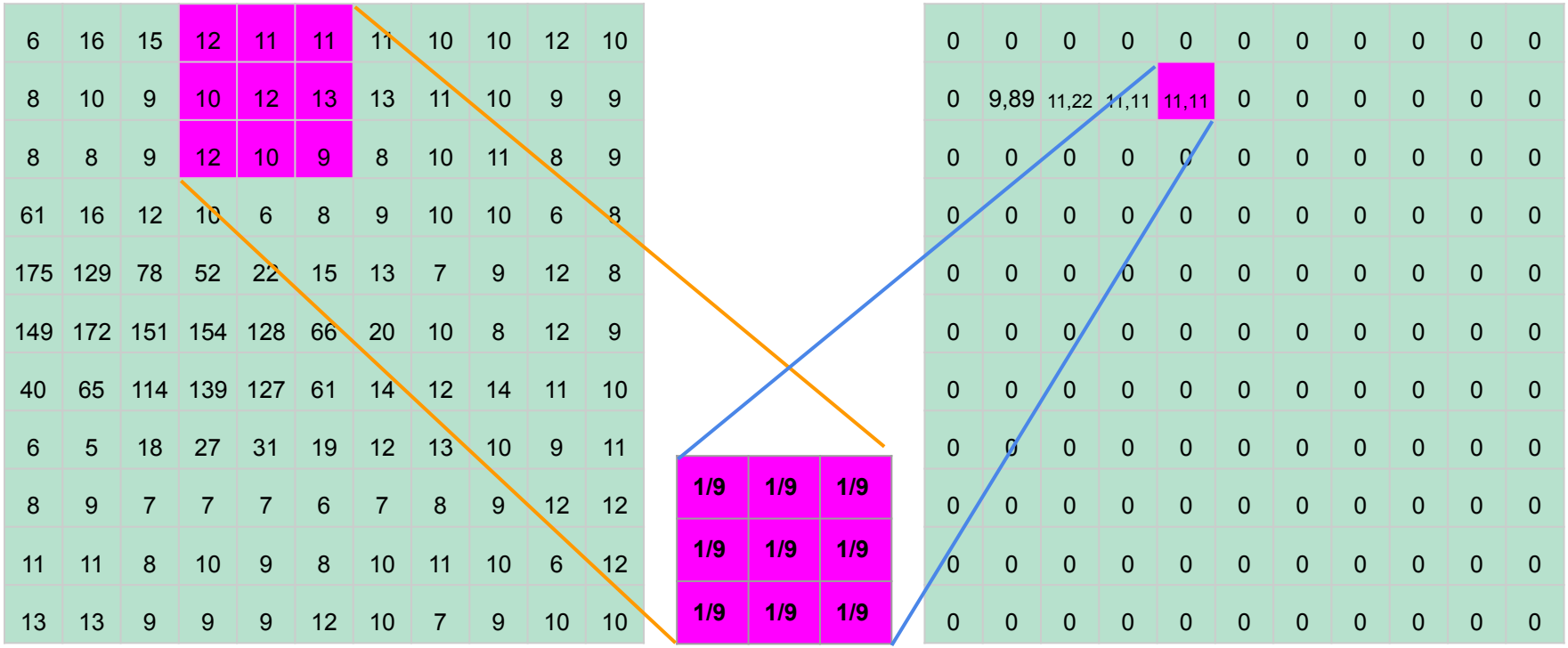


Imagem original - f **Máscara - 9** **Imagem convoluída - f * 9**

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

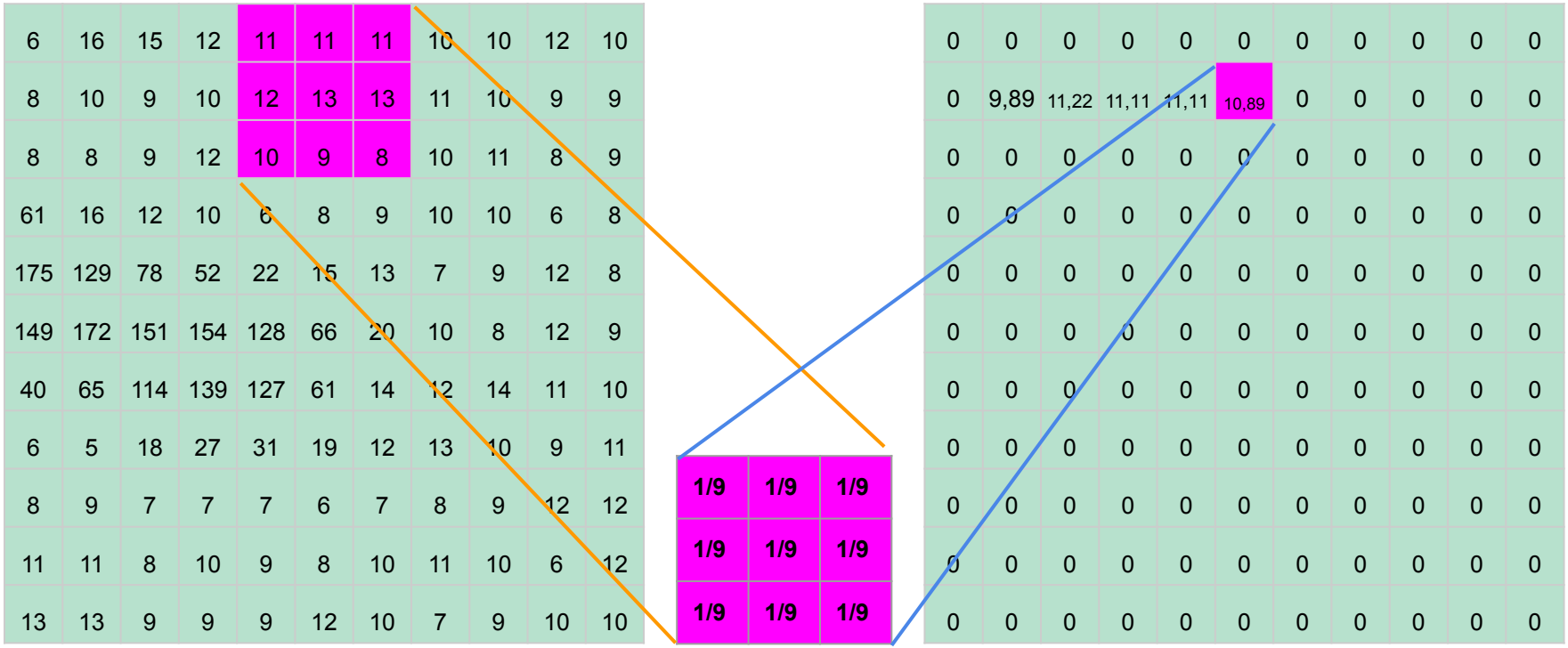


Imagem original - f **Máscara - g** **Imagem convoluída - f * g**

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

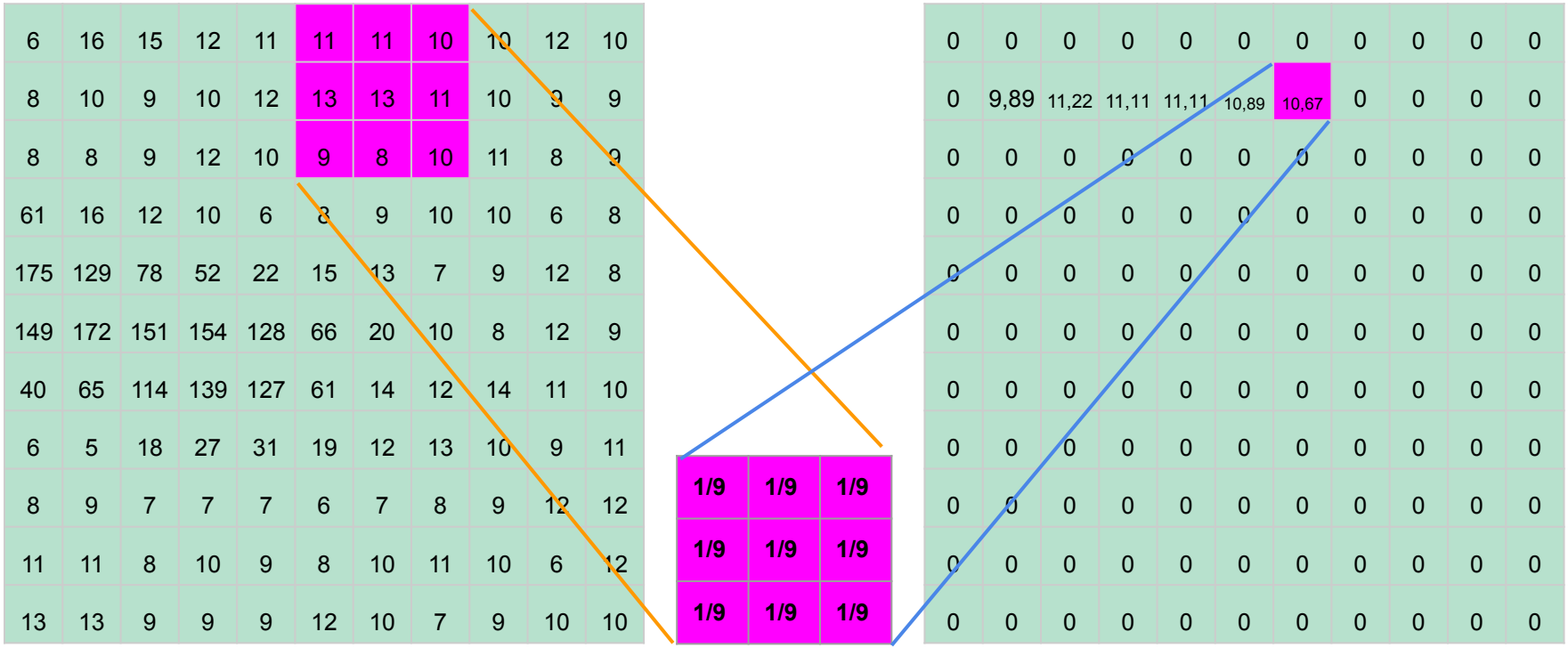


Imagem original - f **Máscara - g** **Imagem convoluída - f * g**

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

6	16	15	12	11	11	11	10	10	12	10
8	10	9	10	12	13	13	11	10	9	9
8	8	9	12	10	9	8	10	11	8	9
61	16	12	10	6	8	9	10	10	6	8
175	129	78	52	22	15	13	7	9	12	8
149	172	151	154	128	66	20	10	8	12	9
40	65	114	139	127	61	14	12	14	11	10
6	5	18	27	31	19	12	13	10	9	11
8	9	7	7	7	6	7	8	9	12	12
11	11	8	10	9	8	10	11	10	6	12
13	13	9	9	9	12	10	7	9	10	10

Imagem original - f

1/9	**1/9**	**1/9**
1/9	**1/9**	**1/9**
1/9	**1/9**	**1/9**

Máscara - g

0	0	0	0	0	0	0	0	0	0	0
0	9,89	11,22	11,11	11,11	10,89	10,67	10,44	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0

Imagem convoluída - f * g

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

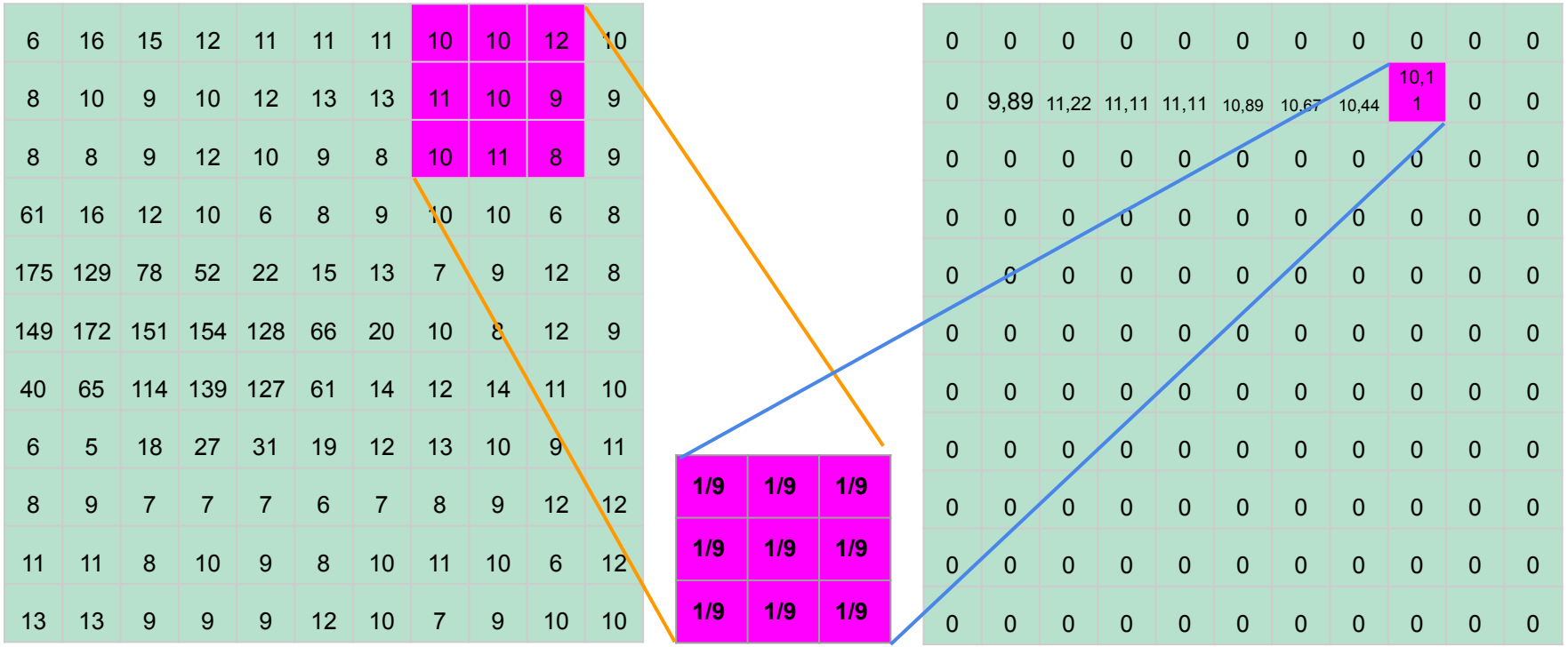


Imagem original - f **Máscara - g** **Imagem convoluída - f * g**

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

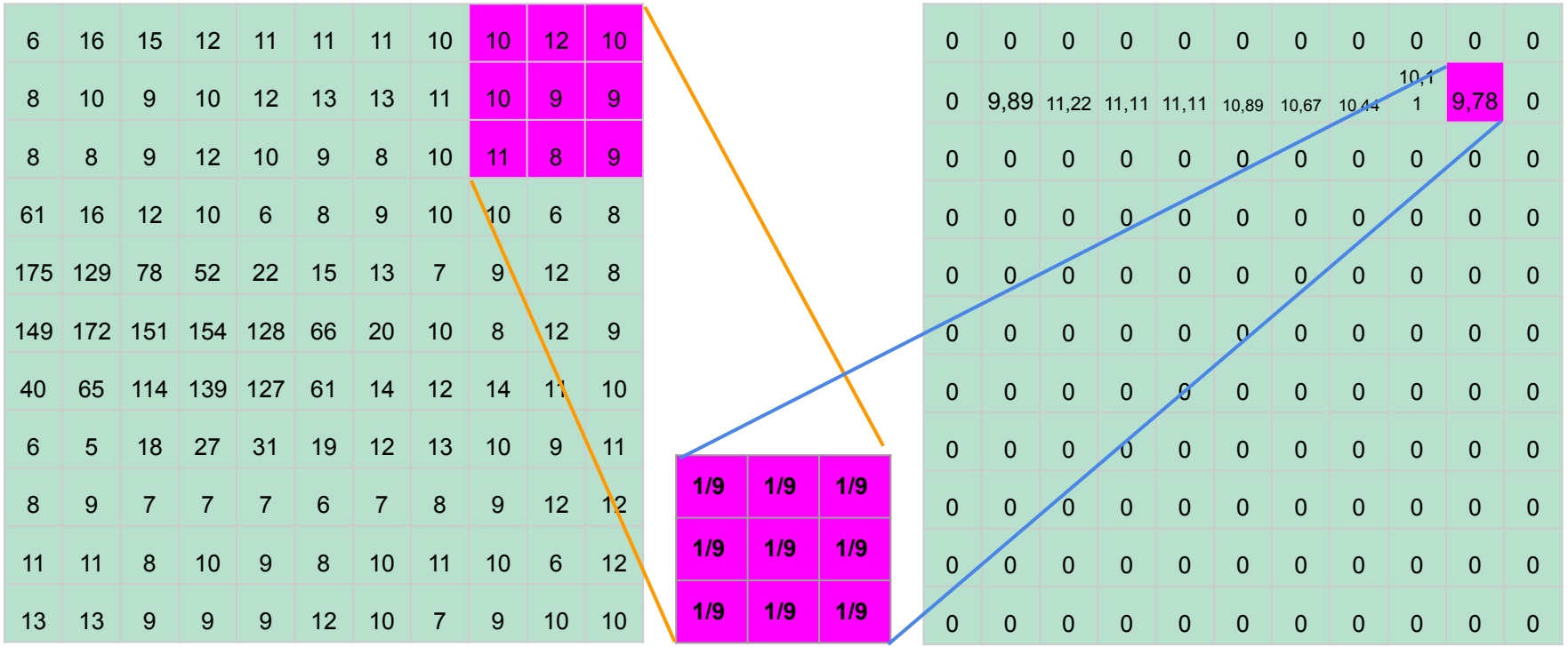


Imagem original - f **Máscara - 9** **Imagem convoluída - f * 9**

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

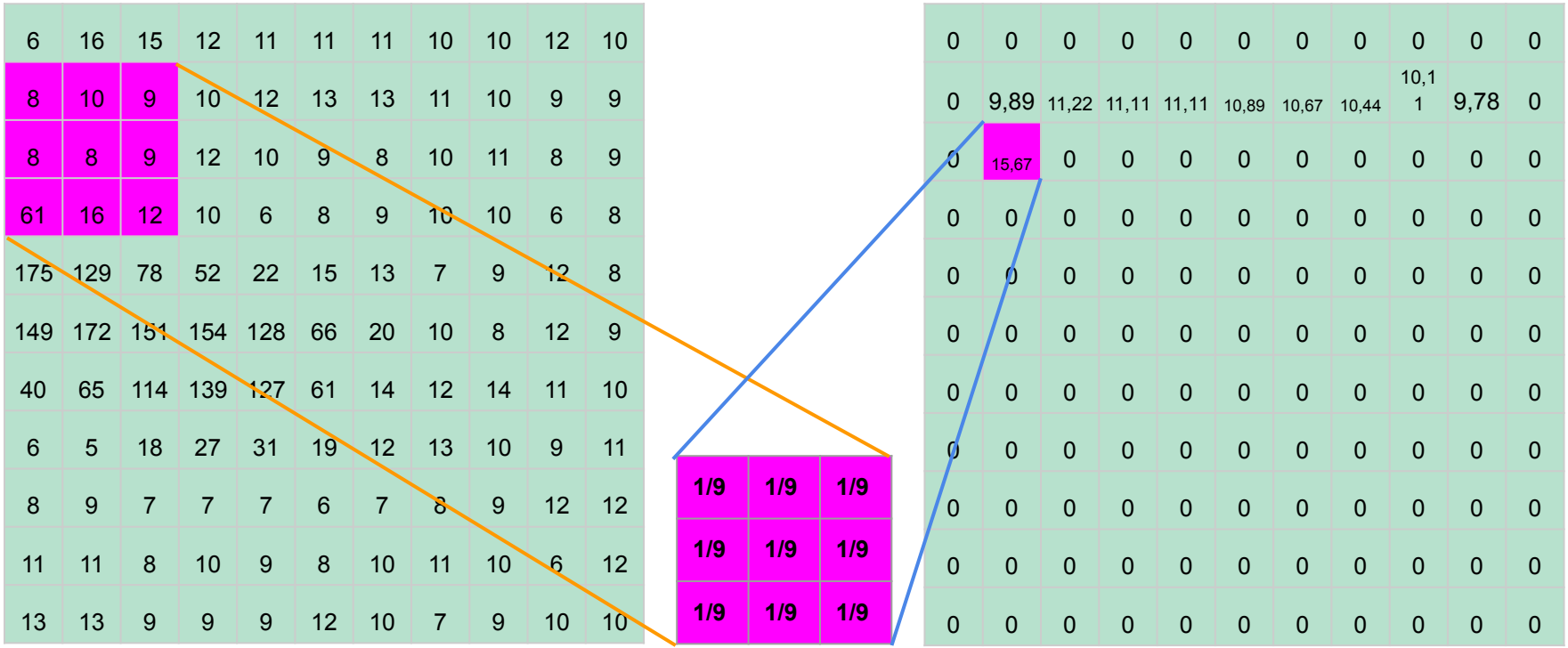


Imagem original - f **Máscara - g** **Imagem convoluída - f * g**

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

6	16	15	12	11	11	11	10	10	12	10
8	10	9	10	12	13	13	11	10	9	9
8	8	9	12	10	9	8	10	11	8	9
61	16	12	10	6	8	9	10	10	6	8
175	129	78	52	22	15	13	7	9	12	8
149	172	151	154	128	66	20	10	8	12	9
40	65	114	139	127	61	14	12	14	11	10
6	5	18	27	31	19	12	13	10	9	11
8	9	7	7	7	6	7	8	9	12	12
11	11	8	10	9	8	10	11	10	6	12
13	13	9	9	9	12	10	7	9	10	10

Imagem original - f

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Máscara - 9

0	0	0	0	0	0	0	0	0	0	0
0	9,89	11,22	11,11	11,11	10,89	10,67	10,44	10,11	9,78	0
0	15,67	10,67	10,00	10,00	9,78	10,11	10,22	9,44	8,89	0
0	55,11	36,22	23,44	16,00	11,11	9,89	9,67	9,22	9,00	0
0	104,78	86,00	68,11	51,22	31,89	17,56	10,67	9,33	9,11	0
0	119,22	117,11	107,22	84,89	51,78	24,22	11,89	10,56	10,33	0
0	80,00	93,89	98,78	83,56	53,11	25,22	12,56	11,00	10,44	0
0	30,22	43,44	53,00	47,11	31,56	16,89	11,00	10,89	10,89	0
0	9,22	11,33	13,78	13,78	12,11	10,44	10,00	9,78	10,11	0
0	9,89	9,22	8,33	8,56	8,67	8,78	9,00	9,11	0	0
0	0	0	0	0	0	0	0	0	0	0

Imagem convoluída - f * 9

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

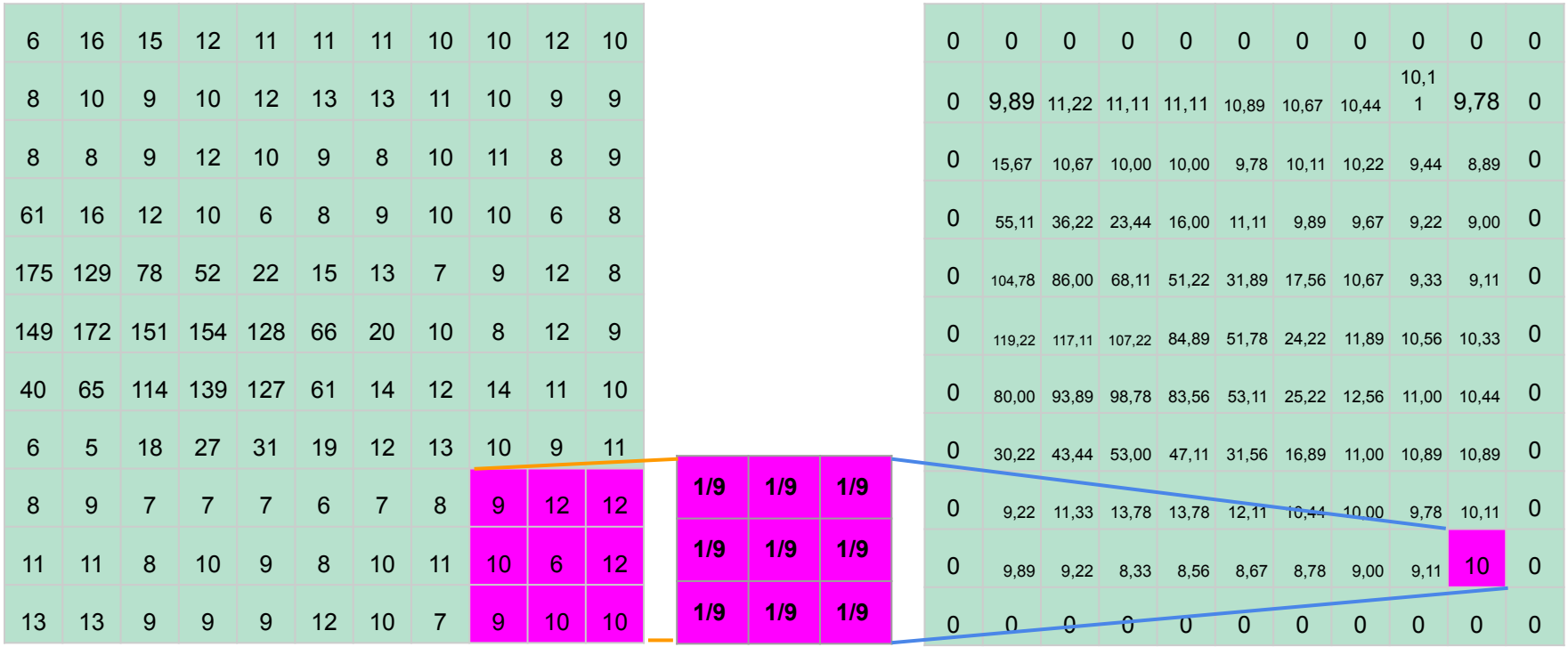


Imagem original - f **Máscara - g** **Imagem convoluída - f * g**

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

6	16	15	12	11	11	11	10	10	12	10
8	10	9	10	12	13	13	11	10	9	9
8	8	9	12	10	9	8	10	11	8	9
61	16	12	10	6	8	9	10	10	6	8
175	129	78	52	22	15	13	7	9	12	8
149	172	151	154	128	66	20	10	8	12	9
40	65	114	139	127	61	14	12	14	11	10
6	5	18	27	31	19	12	13	10	9	11
8	9	7	7	7	6	7	8	9	12	12
11	11	8	10	9	8	10	11	10	6	12
13	13	9	9	9	12	10	7	9	10	10

Imagem original - f

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Máscara - 9

0	0	0	0	0	0	0	0	0	0	0
0	10	11	11	11	11	11	10	10	10	0
0	16	11	10	10	10	10	10	9	9	0
0	55	36	23	16	11	10	10	9	9	0
0	105	86	68	51	32	18	11	9	9	0
0	119	117	107	85	52	24	12	11	10	0
0	80	94	99	84	53	25	13	11	10	0
0	30	43	53	47	32	17	11	11	11	0
0	9	11	14	14	12	10	10	10	10	0
0	10	9	8	9	9	9	9	9	10	0
0	0	0	0	0	0	0	0	0	0	0

Imagem convoluída - f * 9

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtragem convolucional

- De acordo com os valores dos pesos os filtros realizarão diferentes tipos de ações sobre a imagem
  - Suavização
  - Filtragem de ruído
  - Realce de contraste
  - Detecção de bordas
  - Detecção de atributos locais

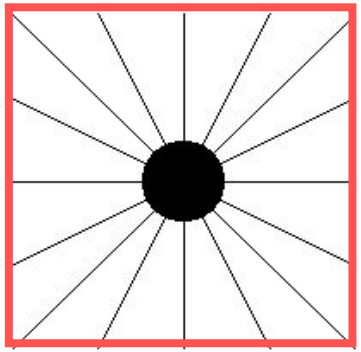
Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtros direcionais

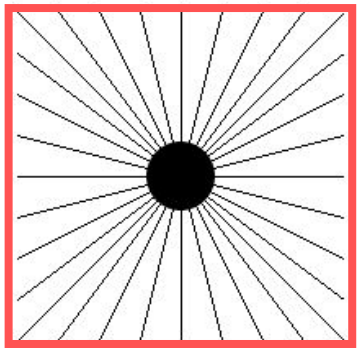
Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtros direcionais

- Exemplo
  - Se aplicarmos a convolução com a máscara K abaixo nas duas imagens ao lado
  - Como você acha que será o resultado?



$$K := \begin{pmatrix} -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \end{pmatrix}$$

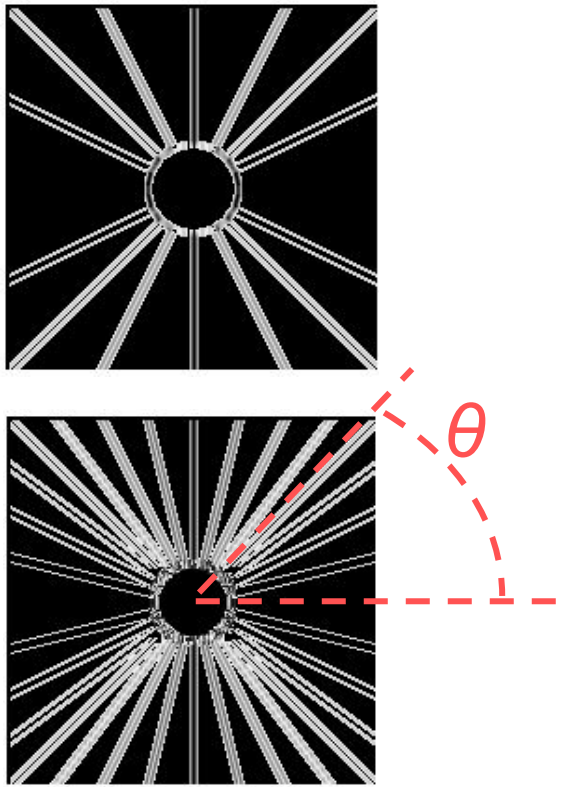


Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtros direcionais

- Resultados:
  - K tem o gradiente na horizontal
    - Detecta bordas orientadas na vertical
  - A resultante têm intensidades variando com a inclinação θ

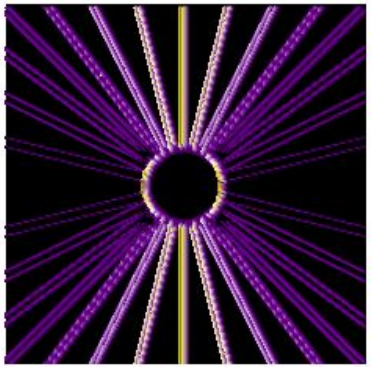
$$K := \begin{pmatrix} -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \end{pmatrix}$$

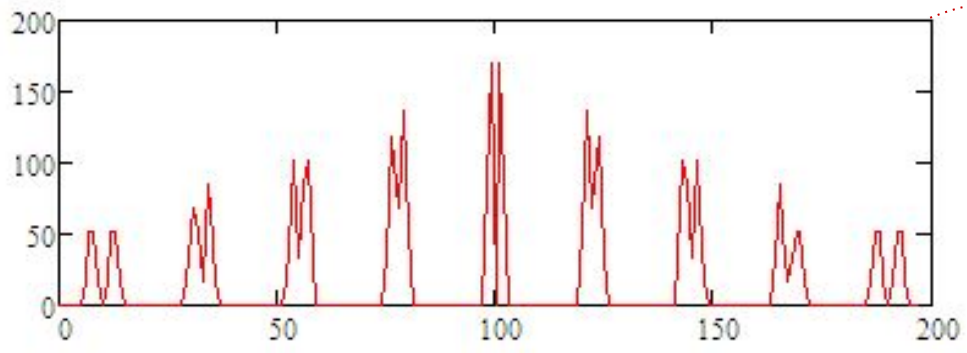


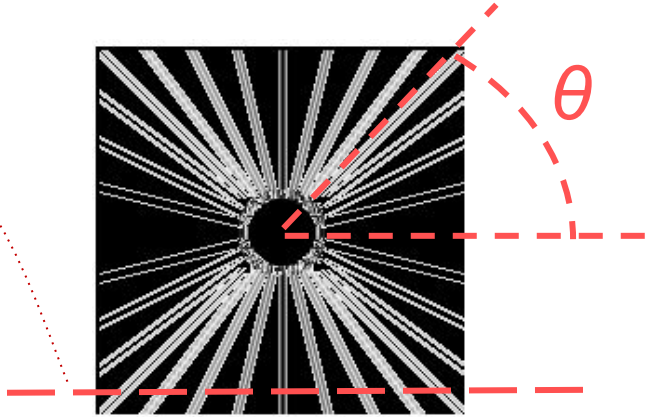
Link do slido de hoje: https://app.sli.do/event/bldcpzub

Filtros direcionais

- Resultados:
  - Exibindo-se as intensidades com cor falsa, notamos mais nitidamente a sua variação com o ângulo θ
  - Façamos um gráfico da intensidade ao longo de uma linha horizontal da imagem







Link do slido de hoje: https://app.sli.do/event/bldcpzub

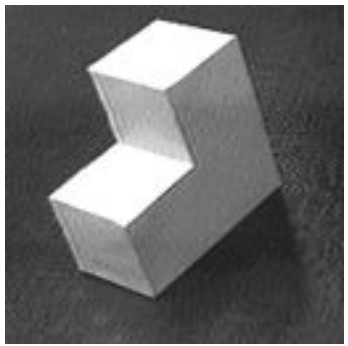
Exemplo: bloco

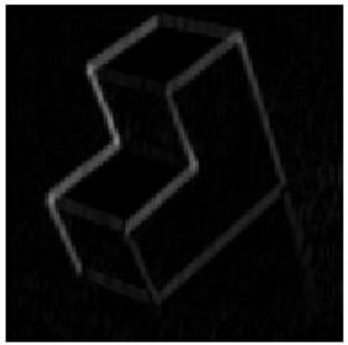
- Aplicando-se o mesmo núcleo à imagem do bloco obtém-se:
- E, após a limiarização:
  - **O limiar seleciona as respostas que são de fato mais significativas**

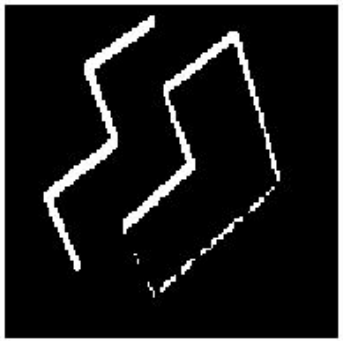
Filtros direcionais

$$K := \begin{pmatrix} -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \end{pmatrix}$$

gradiente







Link do slido de hoje: https://app.sli.do/event/bldcpzub

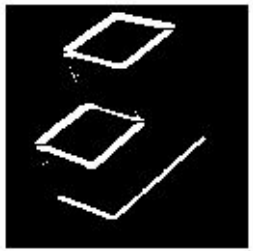
limiarizada

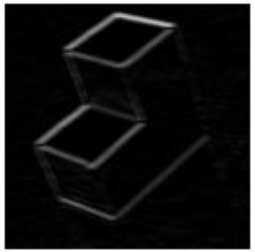
filtrada

máscara

$$\begin{pmatrix} -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \\ -1 & -2 & 0 & 2 & 1 \end{pmatrix}$$

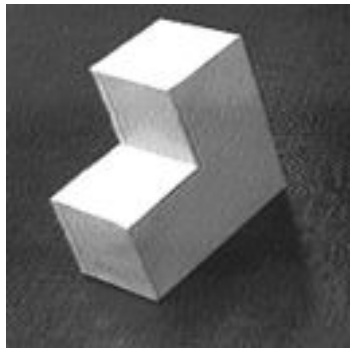
Filtros direcionais

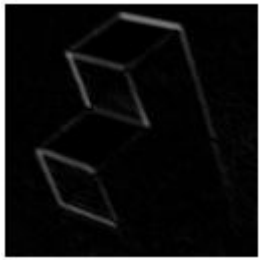




$$\begin{pmatrix} -1 & -1 & -1 & -1 & -1 \\ -2 & -2 & -2 & -2 & -2 \\ 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

original





$$\begin{pmatrix} 0 & -2 & -1 & -1 & -1 \\ 2 & 0 & -2 & -1 & -1 \\ 1 & 2 & 0 & -2 & -1 \\ 1 & 1 & 2 & 0 & -2 \\ 1 & 1 & 1 & 2 & 0 \end{pmatrix}$$

Link do slido de hoje: https://app.sli.do/event/bldcpzub

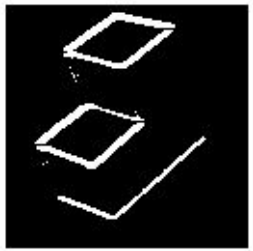
limiarizada

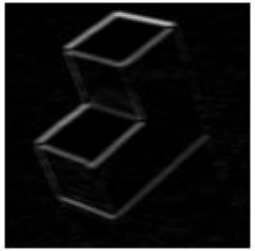
filtrada

máscara

$$\begin{pmatrix} -1 & -1 & -1 & -2 & 0 \\ -1 & -1 & -2 & 0 & 2 \\ -1 & -2 & 0 & 2 & 1 \\ -2 & 0 & 2 & 1 & 1 \\ 0 & 2 & 1 & 1 & 1 \end{pmatrix}$$

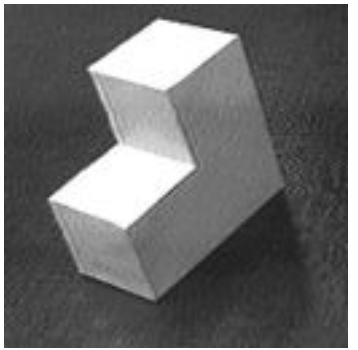
Filtros direcionais

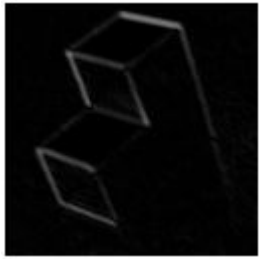




$$\begin{pmatrix} -1 & -1 & -1 & -1 & -1 \\ -2 & -2 & -2 & -2 & -2 \\ 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

original





$$\begin{pmatrix} 0 & -2 & -1 & -1 & -1 \\ 2 & 0 & -2 & -1 & -1 \\ 1 & 2 & 0 & -2 & -1 \\ 1 & 1 & 2 & 0 & -2 \\ 1 & 1 & 1 & 2 & 0 \end{pmatrix}$$

Link do slido de hoje: https://app.sli.do/event/bldcpzub

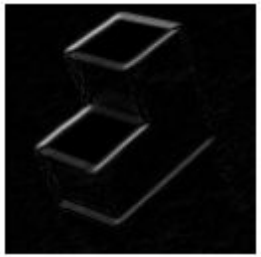
limiarizada

filtrada

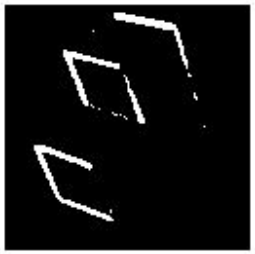
máscara

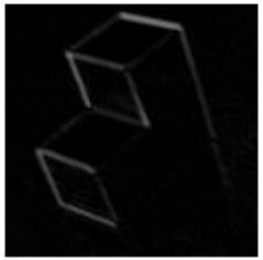
$$\begin{pmatrix} -1 & -1 & -1 & -2 & 0 \\ -1 & -1 & -2 & 0 & 2 \\ -1 & -2 & 0 & 2 & 1 \\ -2 & 0 & 2 & 1 & 1 \\ 0 & 2 & 1 & 1 & 1 \end{pmatrix}$$

Filtros direcionais



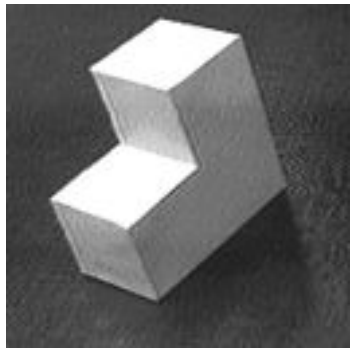
$$\begin{pmatrix} -1 & -1 & -1 & -2 & -2 \\ -1 & -1 & -2 & 0 & 0 \\ -2 & -2 & 0 & 2 & 2 \\ 0 & 0 & 2 & 1 & 1 \\ 2 & 2 & 1 & 1 & 1 \end{pmatrix}$$





$$\begin{pmatrix} 0 & -2 & -1 & -1 & -1 \\ 2 & 0 & -2 & -1 & -1 \\ 1 & 2 & 0 & -2 & -1 \\ 1 & 1 & 2 & 0 & -2 \\ 1 & 1 & 1 & 2 & 0 \end{pmatrix}$$

original



Link do slido de hoje: https://app.sli.do/event/bldcpzub

Detecção de cantos

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Detecção de cantos (*corners*)

Pode ser realizado através de filtros direcionais mais complexos

- A complexidade do filtro está na sua concepção
  - Além de estabelecer a geometria do atributo a ser filtrado (canto) os coeficientes da máscara devem ser balanceados (os valores positivos e negativos devem se equilibrar)
- Em alguns casos é necessário usar vizinhanças maiores, para descrever a geometria a ser filtrada

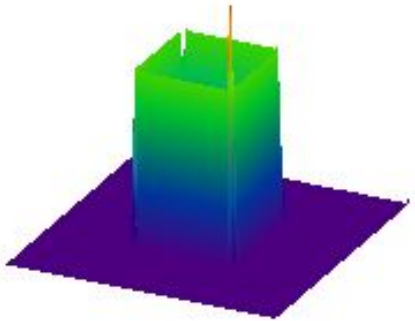
Link do slido de hoje: https://app.sli.do/event/bldcpzub

Exemplo

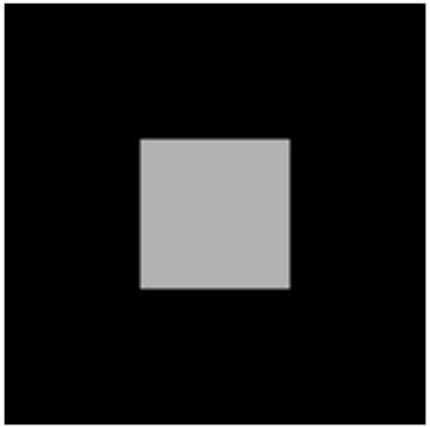
Máscara

$$\begin{pmatrix} -1 & -1 & -1 \\ -1 & 0 & 0 \\ -1 & 0 & 5 \end{pmatrix}$$

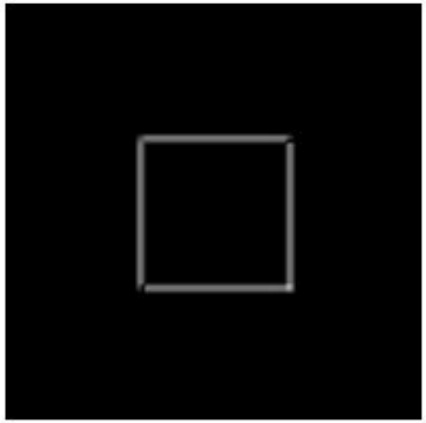
Intensidade Filtrada



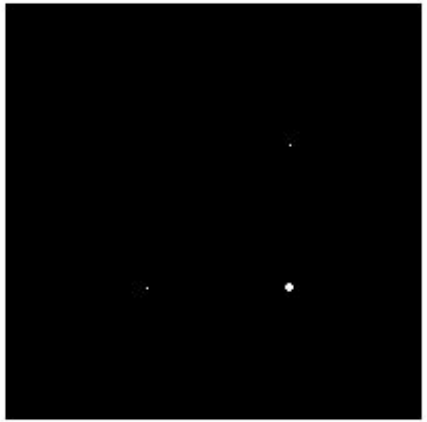
Original



Filtrada



Limiarizada



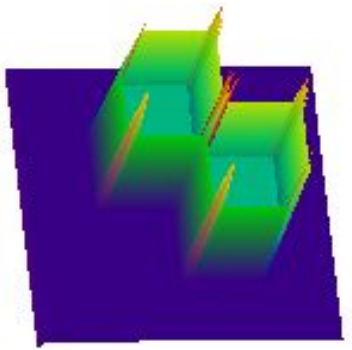
Link do slido de hoje: https://app.sli.do/event/bldcpzub

Exemplo

Máscara

$$\begin{pmatrix} -1 & -1 & 1 & 0 & 8.5 \\ -1 & -1 & -1 & 0 & 0 \\ -1 & -1 & -1 & -1 & -1 \\ 0 & 0 & -1 & -1 & 1 \\ 8.5 & 0 & -1 & -1 & -1 \end{pmatrix}$$

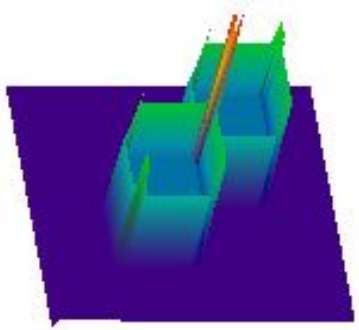
Intensidades Filtradas



Originais

Filtradas

Limiarizadas



Link do slido de hoje: https://app.sli.do/event/bldcpzub

Um exemplo completo

Identificação “online” de caracteres manuscritos

Fonte: https://jackschaedler.github.io/handwriting-recognition/

Link do slido de hoje: https://app.sli.do/event/bldcpzub

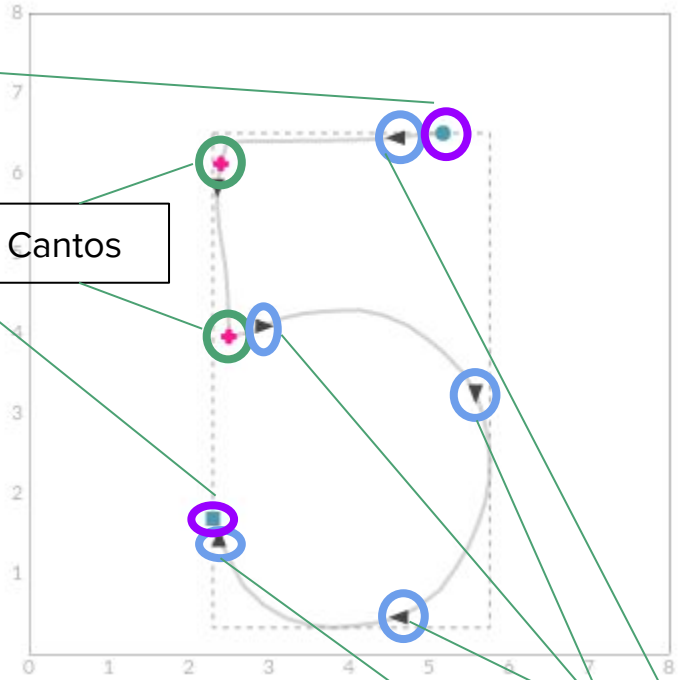
Identificação via atributos locais

- A identificação de caracteres manuscritos é um dos problemas mais antigos e difíceis em Visão Computacional.
- Em 1966, um grupo de cientistas que trabalhavam para a RAND Corporation publicaram um artigo interno sobre um reconhecedor de caracteres para um dispositivo do tipo “tablet” baseado em atributos locais como **cantos**, **curvatura**, **posicionamento**, **razão proporcional**, **tamanho** e **terminações**.
- O classificador da RAND era baseado em regras que analisavam esses atributos locais e classificavam os caracteres.

Fonte: http://www.rand.org/pubs/research_memoranda/RM5016.html

Link do slido de hoje: https://app.sli.do/event/bldcpzub

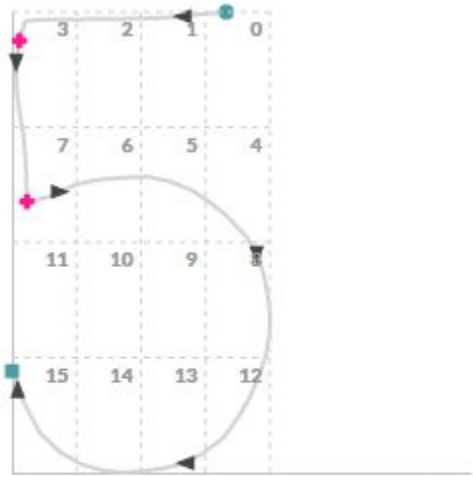
Identificação via atributos locais - exemplos



Terminações

Cantos

Curvaturas



Center: (4.03, 3.41)
Stroke Start: 0
Stroke End: 15
Directions: ◄ ▼ ► ▼ ◄ ▲
Corners: 3,7

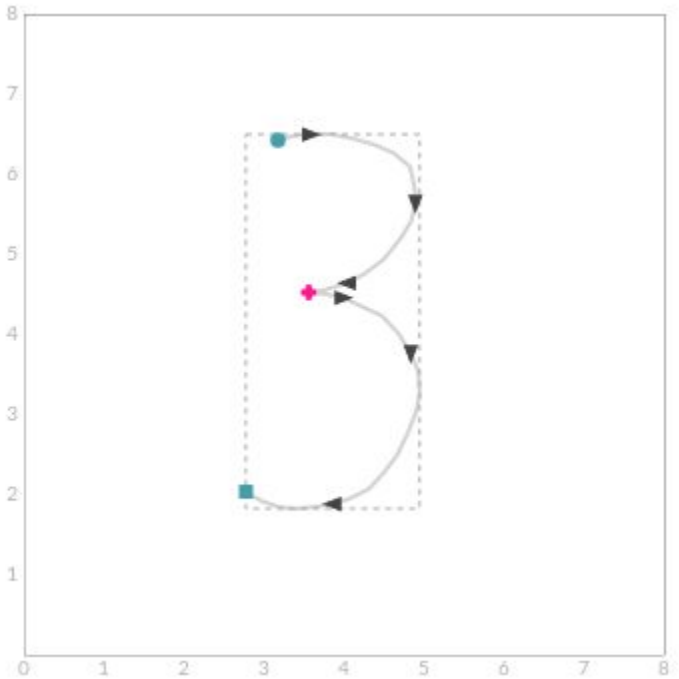
Width: 3.46
Height: 6.17
Aspect Ratio: 1.78

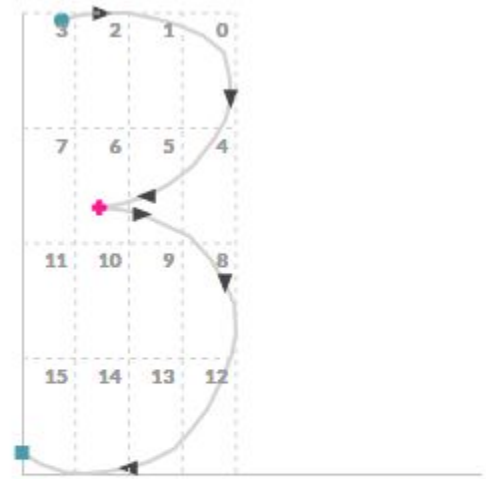
Tamanho

Razão de Proporção

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Identificação via atributos locais - exemplos



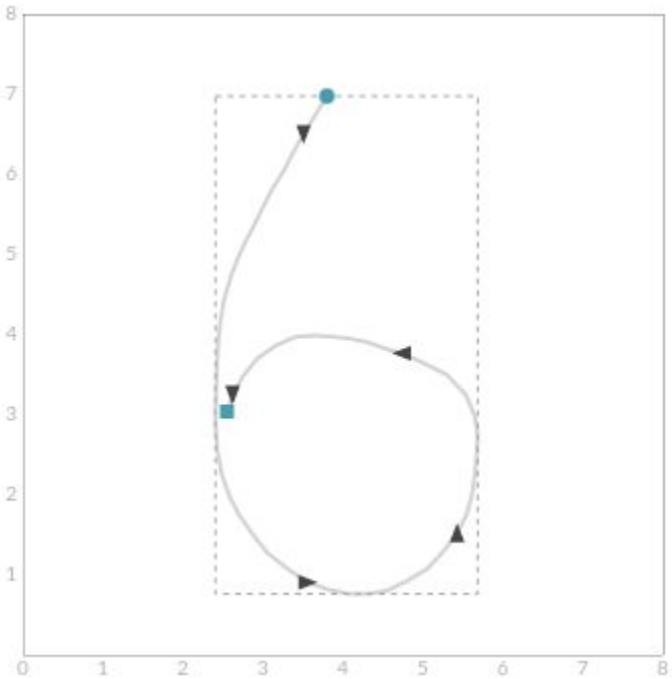


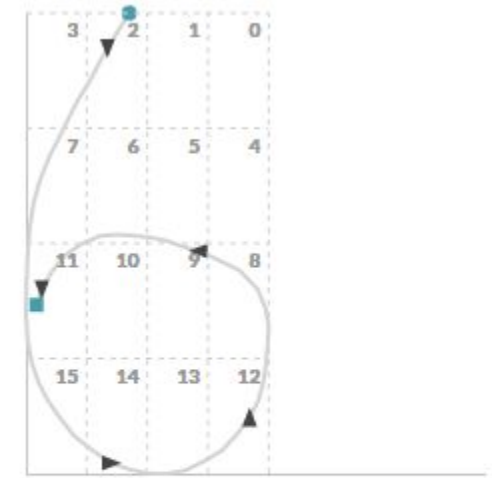
Center: (3.86, 4.16)
Stroke Start: 3
Stroke End: 15
Directions: ► ▼ ◄ ► ▼ ◄
Corners: 6

Width: 2.17
Height: 4.67
Aspect Ratio: 2.15

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Identificação via atributos locais - exemplos



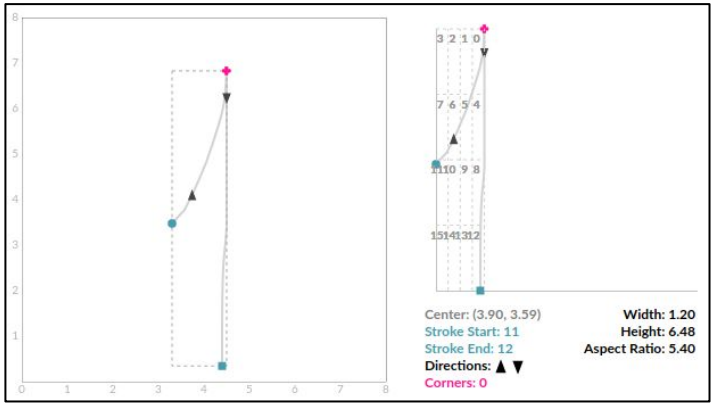


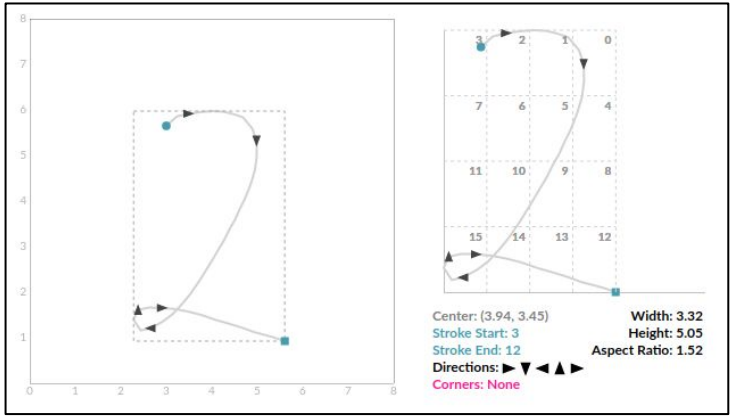
Center: (4.05, 3.87)
Stroke Start: 2
Stroke End: 11
Directions: ▼ ► ▲ ◄ ▼
Corners: None

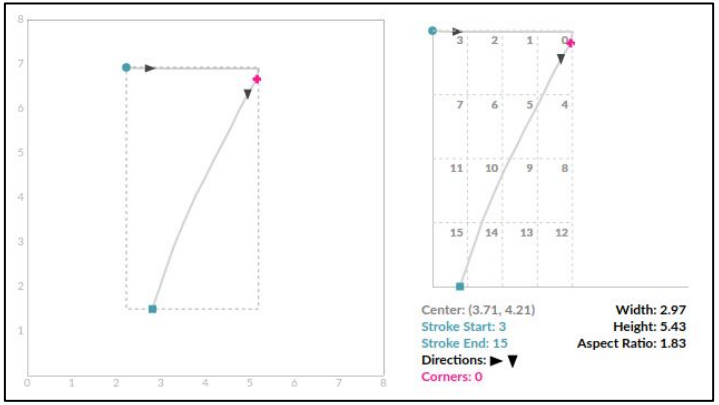
Width: 3.28
Height: 6.21
Aspect Ratio: 1.90

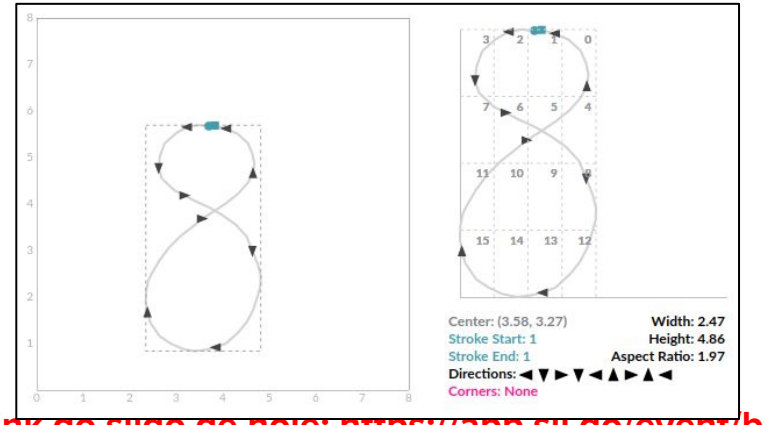
Link do slido de hoje: https://app.sli.do/event/bldcpzub

Identificação via atributos locais - exemplos









Link do slido de hoje: https://app.sli.do/event/bldcpzub

Aprendizado supervisionado

Fonte das imagens: Pattern Classification - Stork, Hart e Duda

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Classificadores supervisionados

- A construção de classificadores depende da escolha de bons atributos
- Depende da quantidade de exemplos de treinamento (para a construção do classificador)
- Depende da qualidade desses exemplos, isto é., o quão fiéis os exemplos são em relação a distribuição de sua população

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Classificadores supervisionados

- Muitas vezes temos mais atributos que são necessários
- Nem sempre é fácil escolher e combinar bons atributos
- Quanto mais atributos, em geral, mais fácil encontrar uma separação natural das classes
- Porém, perde-se robustez. Também com o aumento da complexidade do operador

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Classificadores supervisionados

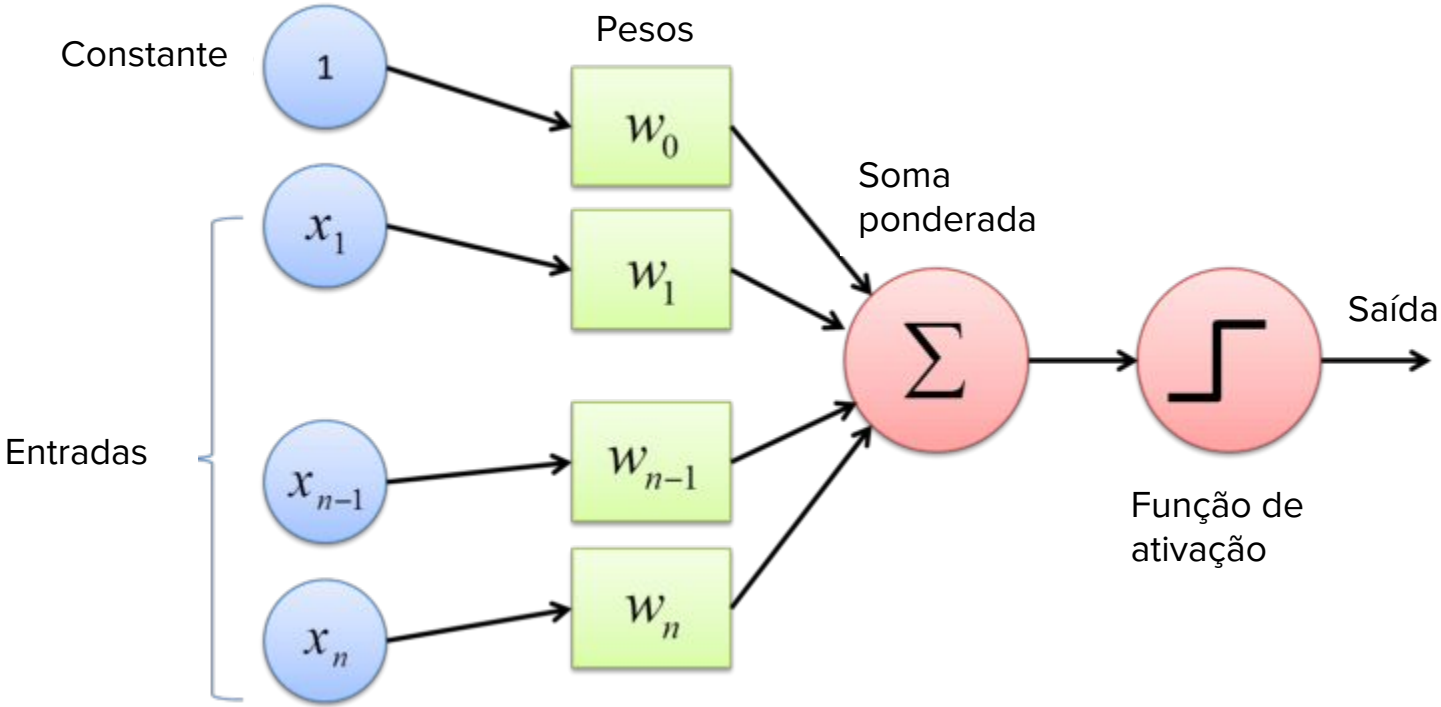
- Quando a complexidade aumenta demais, o classificador fica ajustado demais aos dados
- Em inglês, temos o famoso problema de “overfitting”
- Nesse caso, o erro nos dados de treinamento é zero, ou próximo
- Os erros nos dados de teste podem ser grandes

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Redes neurais

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Redes neurais - Perceptron

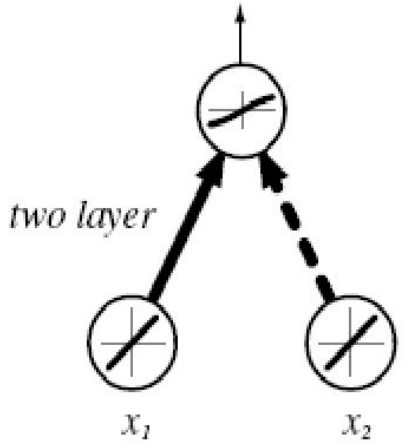


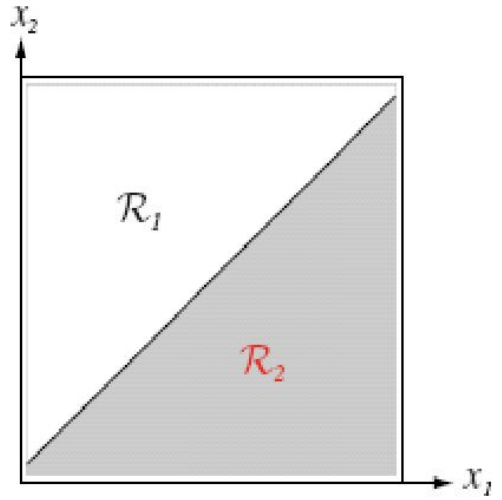
Fonte: https://towardsdatascience.com/what-the-hell-is-perceptron-626217814f53

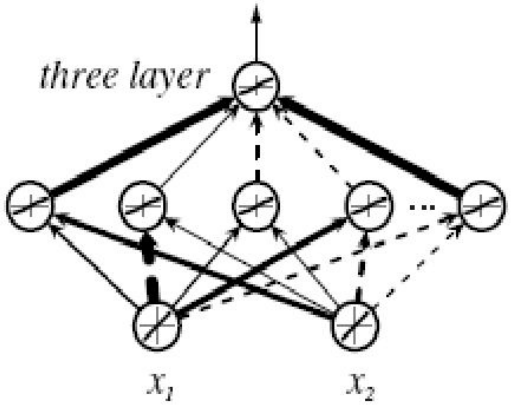
Link do slido de hoje: https://app.sli.do/event/bldcpzub

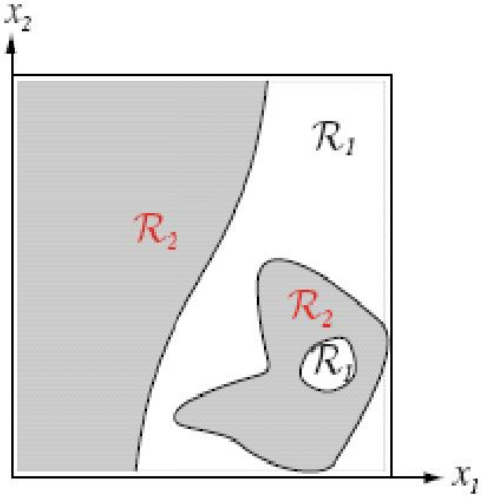
Perceptron

- A soma ponderada feita pelo perceptron nada mais é que um plano num espaço multidimensional, ou hiperplano.
- A função de ativação é usada, normalmente, para atribuir rótulos para os exemplos. Nos dois casos ao lado, R1 e R2 são regiões onde todos os pontos têm rótulos do mesmo valor em cada região.
- Quando combinamos dois, ou mais perceptrons, podemos criar regiões mais complexas.









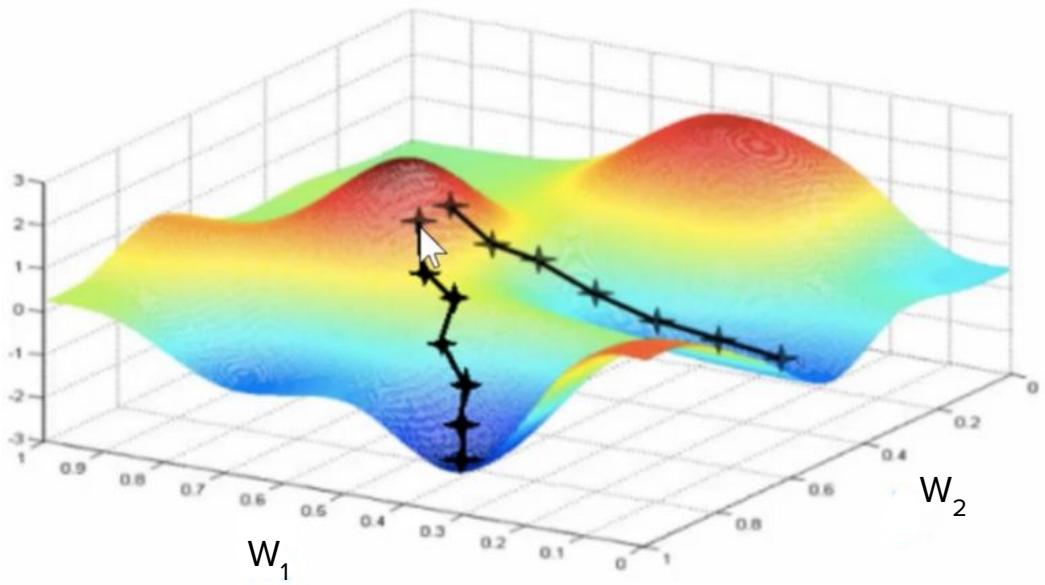
Otimização/treinamento de redes neurais

- O treinamento de redes neurais de várias camadas é feito por um algoritmo conhecido como **backpropagation** que tenta minimizar uma **função de perda**:

$$E(\mathbf{w}) = \sum_{j=1}^{N} \left(y_j - f_{\mathbf{w}}(\mathbf{x}_j) \right)^2$$

- Os pesos são atualizados segundo a fórmula:

$$\mathbf{w} \leftarrow \mathbf{w} - \alpha \frac{\partial E}{\partial \mathbf{w}}$$



Fonte: Jeremy Howard /@fast.ai

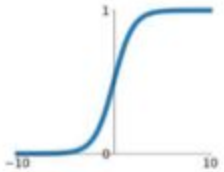
Link do slido de hoje: https://app.sli.do/event/bldcpzub

Funções de ativação

Activation Functions

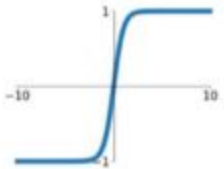
Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



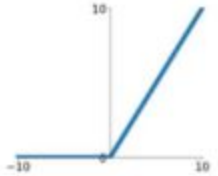
tanh

$$\tanh(x)$$



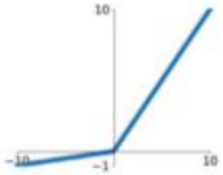
ReLU

$$\max(0, x)$$



Leaky ReLU

$$\max(0.1x, x)$$

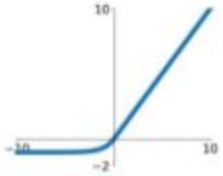


Maxout

$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



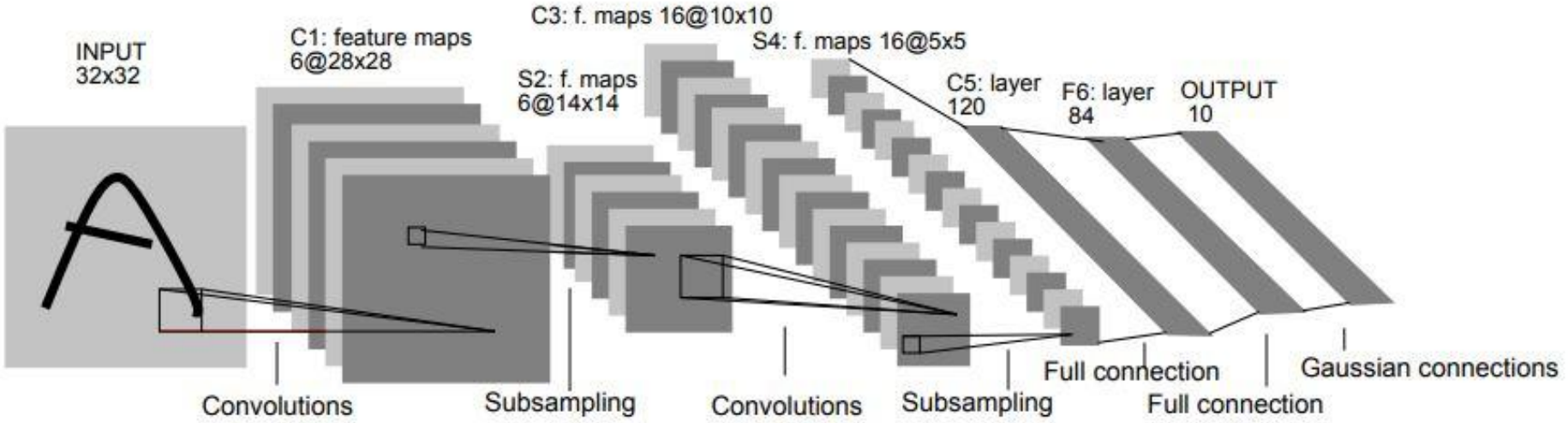
Fonte: https://towardsdatascience.com/complete-guide-of-activation-functions-34076e95d044

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Das redes neurais ao deep learning

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Rede neural convolucional - LeNet 5 - 1990



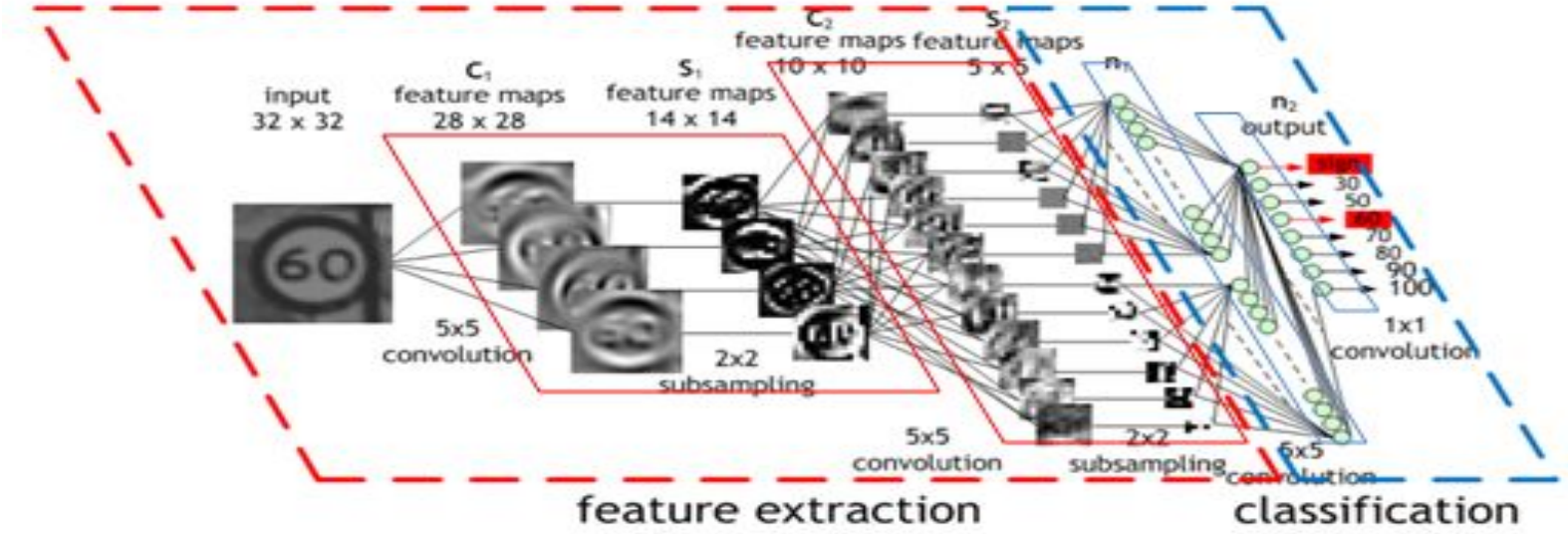
Principais técnicas:

- “average pooling” - espécie de filtro da média, mas com mudança de resolução;
- função de ativação - sigmoid ou tangente hiperbólica (tanh);
- camadas completamente conectadas no final;
- treinado no conjunto MNIST, um conjunto de dígitos com 60K exemplos.

Fonte: http://yann.lecun.com/exdb/publis/pdf/lecun-01a.pdf

Link do slido de hoje: https://app.sli.do/event/bldcpzub

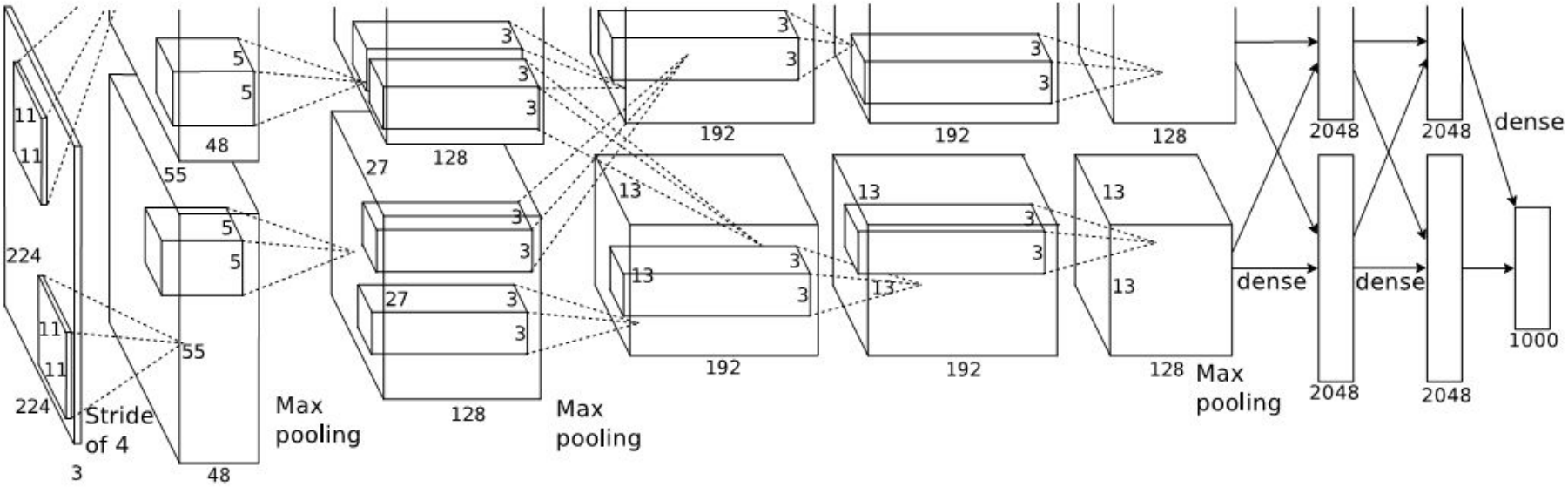
Rede neural convolucional



Fonte: Maurice Peemen - http://parse.ele.tue.nl/mpeemen

Link do slido de hoje: https://app.sli.do/event/bidcpzub

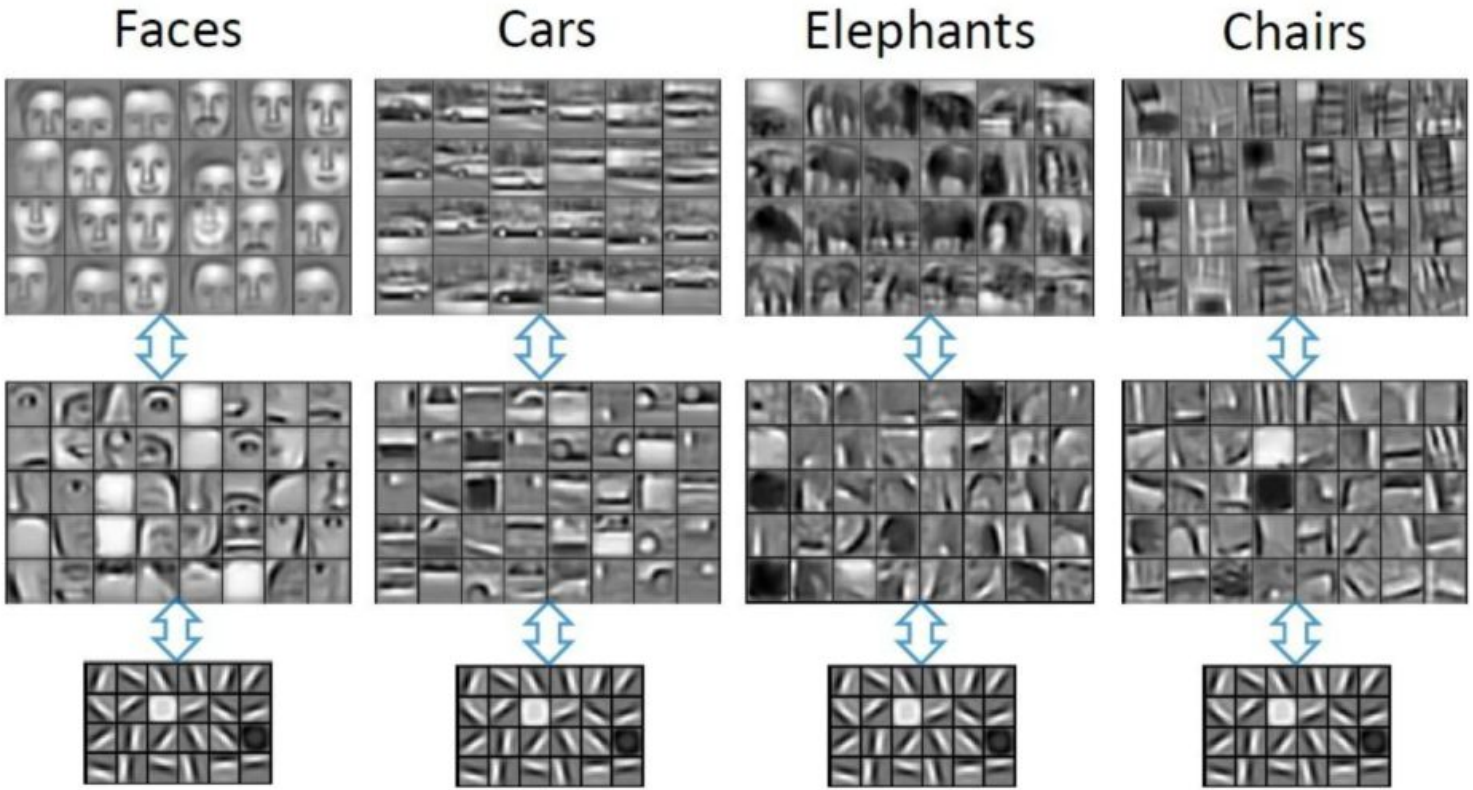
Alex Net - 2012



- Primeira rede CNN a ganhar a competição ImageNet (> 14 milhões de imagens, 20k categorias)
- Cinco camadas convolucionais + três camadas completamente conectadas + classificador softmax (generalização da regressão logística);
- Função de ativação ReLU, “dropout” e aumentação de dados;
- 61 milhões de parâmetros

Link do slido de hoje: https://app.sli.do/event/bldcpzub

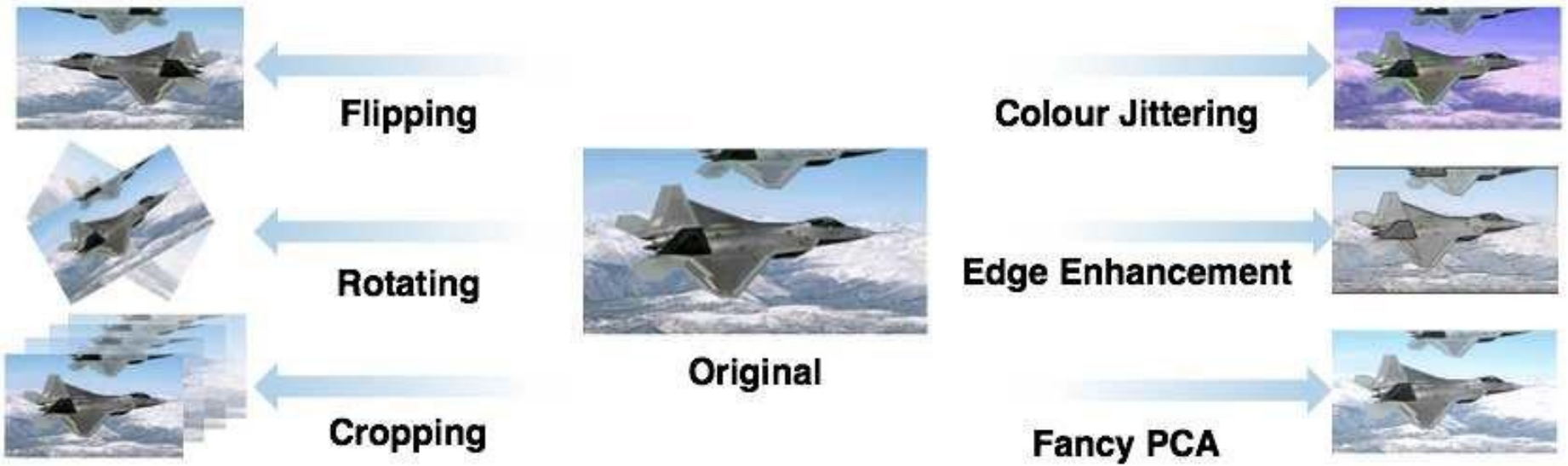
Hierarquia semântica



Fonte: Atlas Wang

Link do slido de hoje: https://app.sli.do/event/bidcpzub

Aumentação de dados



Fonte: Improving Deep Learning using Generic Data Augmentation - Luke Taylor, Geoff Nitschke

Link do slido de hoje: https://app.slido.com/event/blacpzub

VGG Net - 2014

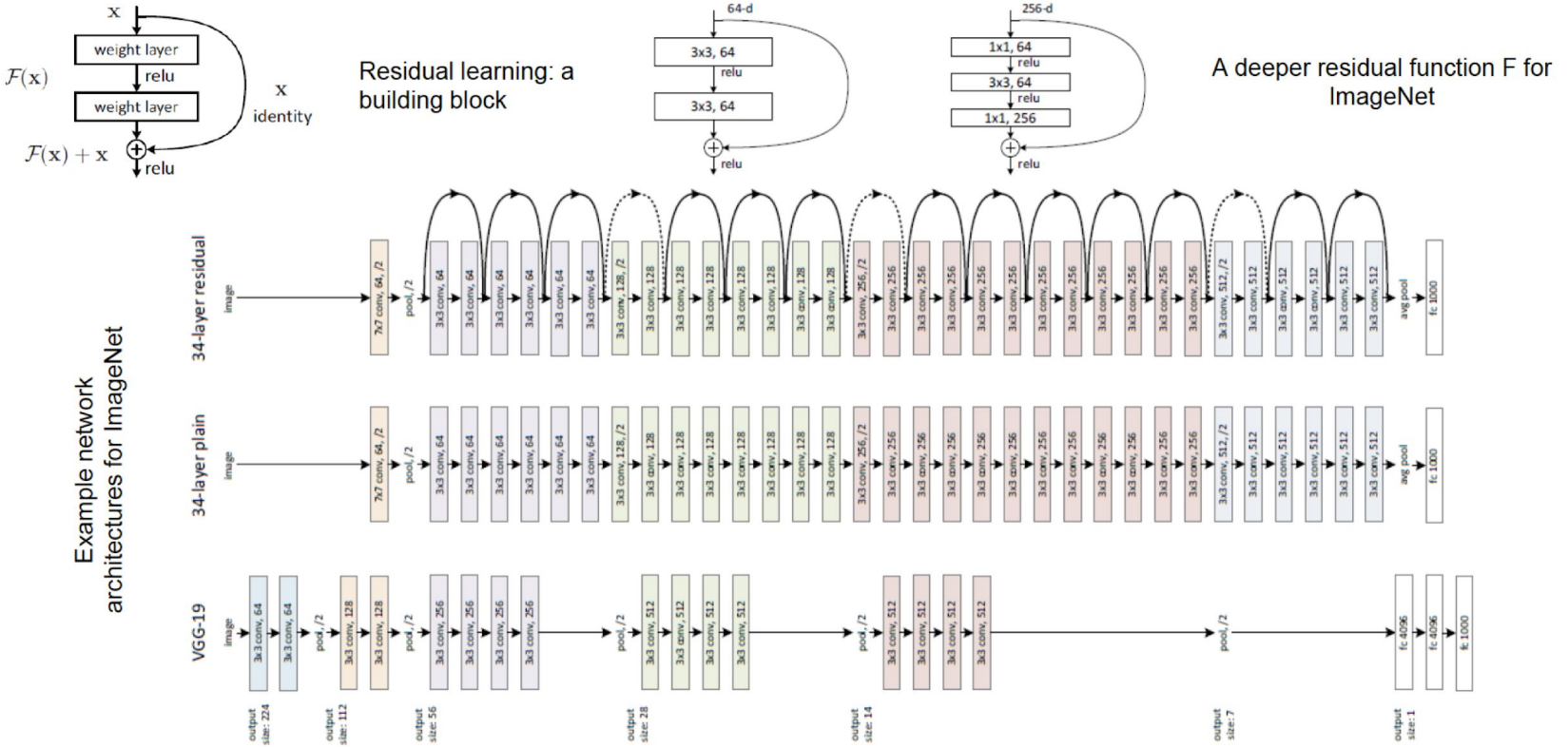
ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

- Aumenta o número de camadas de 8 para 11 até 19
- Diminui o tamanho das máscaras de convolução
- As versões D e E são usadas para muitas tarefas e são conhecidas como VGG 16 e VGG 19
- A VGG 16 tem aproximadamente 138 milhões de parâmetros e a VGG 19 quase 143 milhões

Fonte da tabela: Very Deep Convolutional Networks for Large-Scale Image Recognition - K. Simonyan e A. Zisserman

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Res Net - 2015



Diferenciais: Atinge o patamar humano para classificação no ImageNet; usa a ideia de conexão saltada para mapas residuais; mais de 1000 camadas.

Link do slido de hoje: https://app.sli.do/event/bkdqpzub

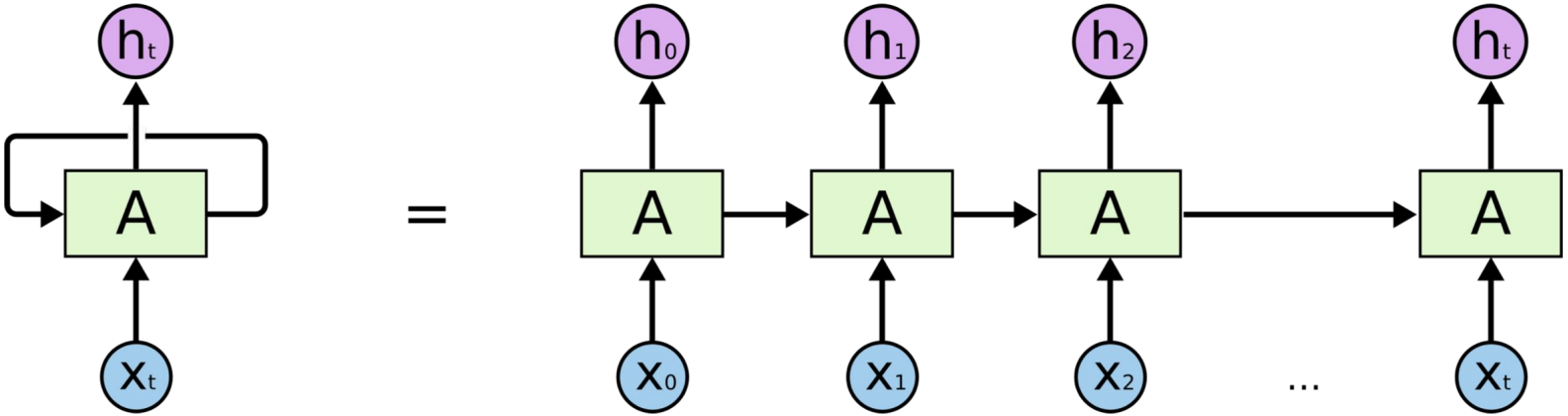
Fonte: Atlas Wang

Tarefa: reconhecimento de objetos/imagens

- Embora as redes convolucionais sejam do final da década de 80, apenas em 2012/2013 elas ganham popularidade
- Em cerca de três/quatro anos, as redes convolucionais alcançam o nível do ser humano para o conjunto ImageNet!
- A quantidade de pesquisa nesses três/quatro anos foi muito grande e diversas novas ideias surgiram, ou ressurgiram com nova roupagem
- Depois da ResNet, apareceram muitas outras redes para reconhecimento de objetos/imagens: Wide ResNet (2016), DenseNet (2017), SeNet (2017) etc.

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Recurrent Neural Network

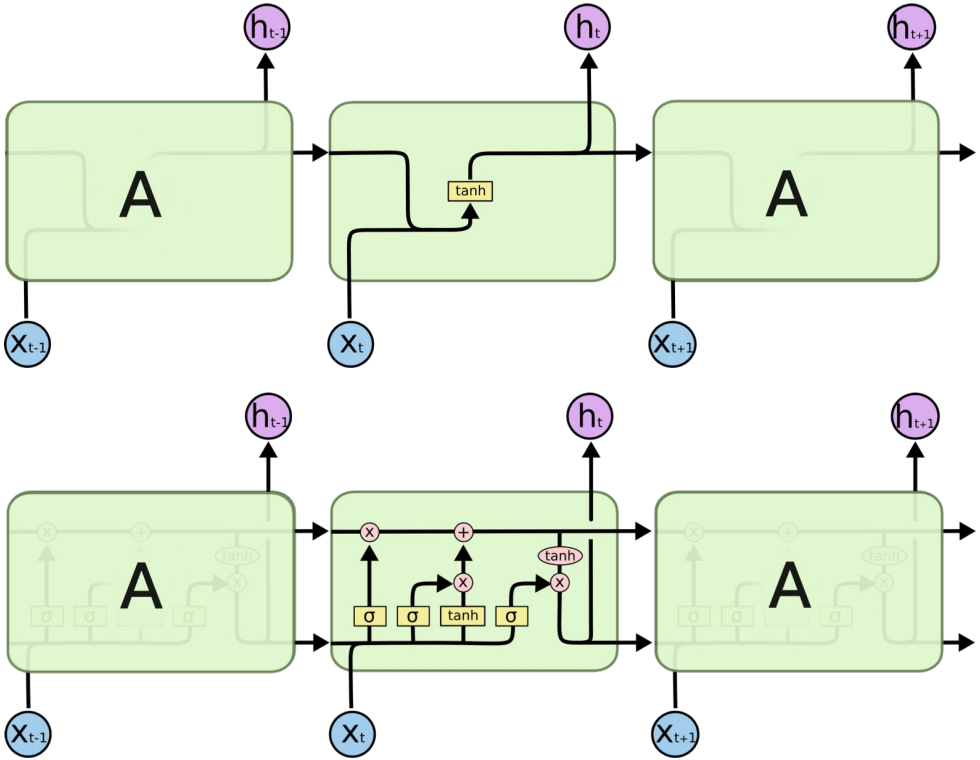


- RNN é um rede com “loops” dentro delas
- Para otimizá-las, usa-se o Backpropagation Through Time (BPTT). Os parâmetros são compartilhados por todos os passos temporais na rede (o gradiente em cada saída depende dos cálculos de todos os passos).
- Problema dos gradientes desaparecendo/explodindo com o comprimento

Fonte: RNN/LSTM: “Understanding LSTM Networks”: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Link do slido de hoje: https://app.sli.do/event/bidcpzub

RNN e LSTM



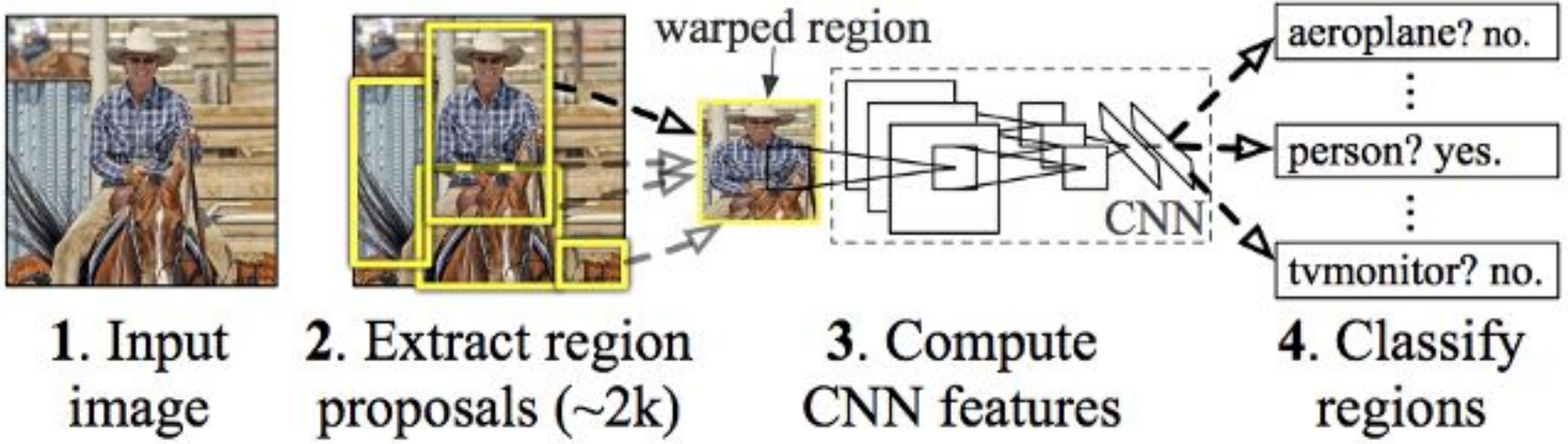
- Long Short Term Memory (LSTM)

  resolve o problema dos gradientes com um mecanismo de portas (gates) que controlam a “memória’
- Mecanismo parecido com os atalhos da ResNet
- LSTM faz o mesmo que a RNN, de um jeito diferente

Fonte: RNN/LSTM: “Understanding LSTM Networks”: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Link do slide de hoje: https://app.sli.do/event/bidcpzub

Regiões com atributos CNN - R-CNN - 2014

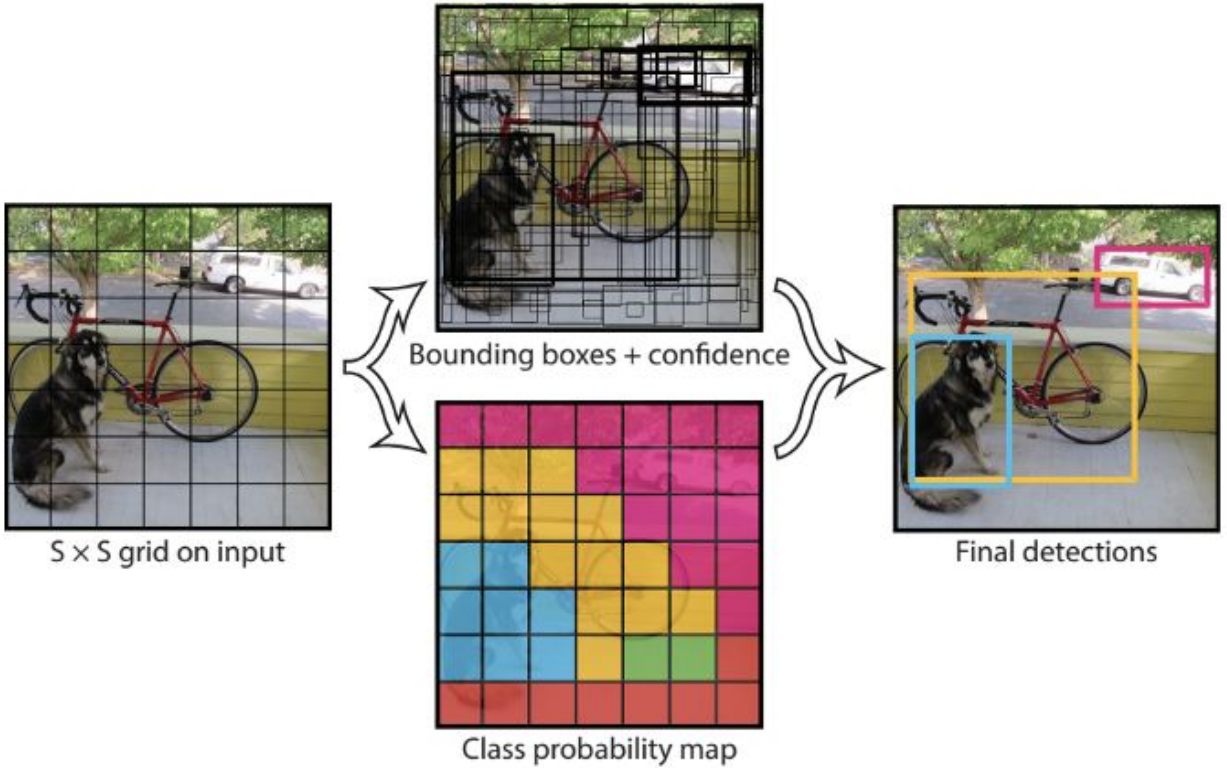


Problemas: Tem que analisar muitas regiões (~2000) por isso é lento.
O mesmo autor propôs uma outra rede que melhorava a primeira: Fast R-CNN

Fonte: Girshick et. al. https://arxiv.org/pdf/1311.2524.pdf

Link do slido de hoje: https://app.sli.do/event/bldcpzub

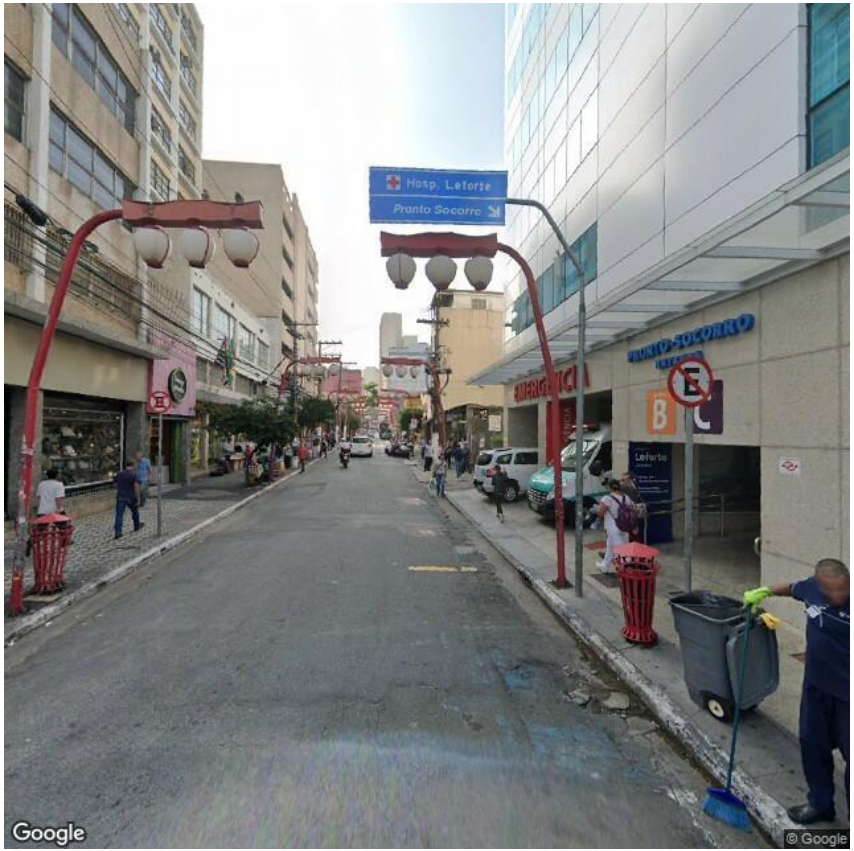
You only look once - YOLO - 2016

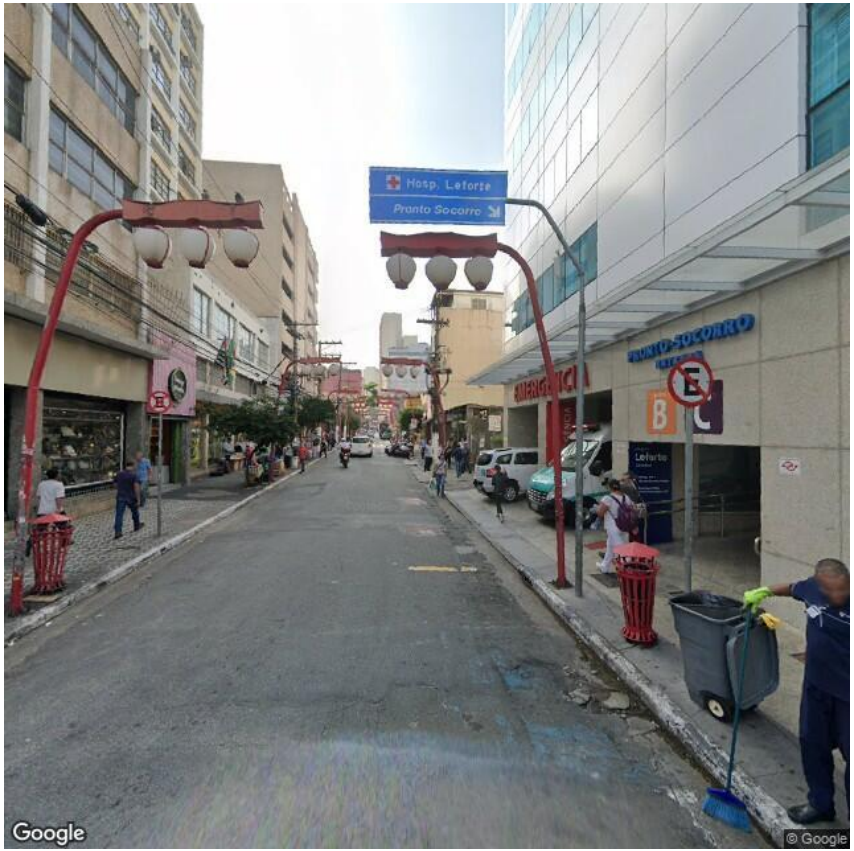


Fonte: Redmon et al: https://arxiv.org/pdf/1506.02640v5.pdf

Link do slido de hoje: https://app.sli.do/event/bldcpzub

YOLO v3

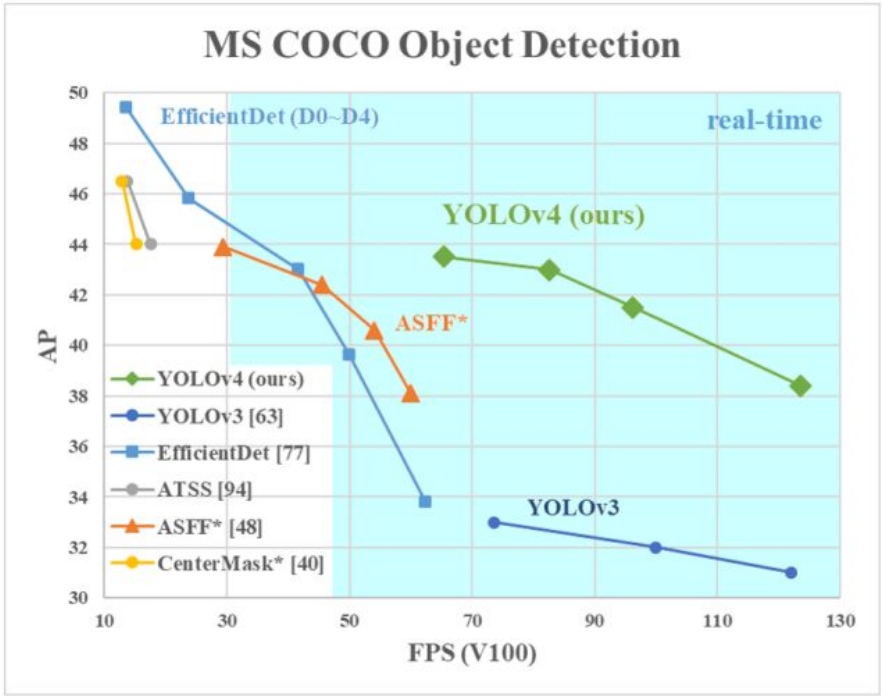




Software: https://pjreddie.com/darknet/yolo/

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Tarefa: localização e reconhecimento de objetos



YOLO v5 e variantes - max 140 FPS

Fonte: https://blog.roboflow.ai/yolov5-is-here/

https://blog.roboflow.ai/yolov5-improvements-and-evaluation/

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Outras razões do sucesso

- GPU
- Grande quantidade de dados rotulados
- Python
- Competições
- Software Livre

Link do slido de hoje: https://app.sli.do/event/bldcpzub

GPU – a serviço dos jogos

- NVIDIA – GeForce 256 - 1999
- Considerada a primeira GPU - 256MB!
- 10 milhões de polígonos por segundo!

Link do slido de hoje: https://app.sli.do/event/bldcpzub

GPGPU – a serviço dos jogos e ciência

- NVIDIA – GeForce GT 8800 – 2007 – 256M/512M/1024M
- 4 anos de desenvolvimento, quase US$500M, 800M transistores
- Melhoramentos significativos na arquitetura – 128 processadores

Link do slido de hoje: https://app.sli.do/event/bldcpzub

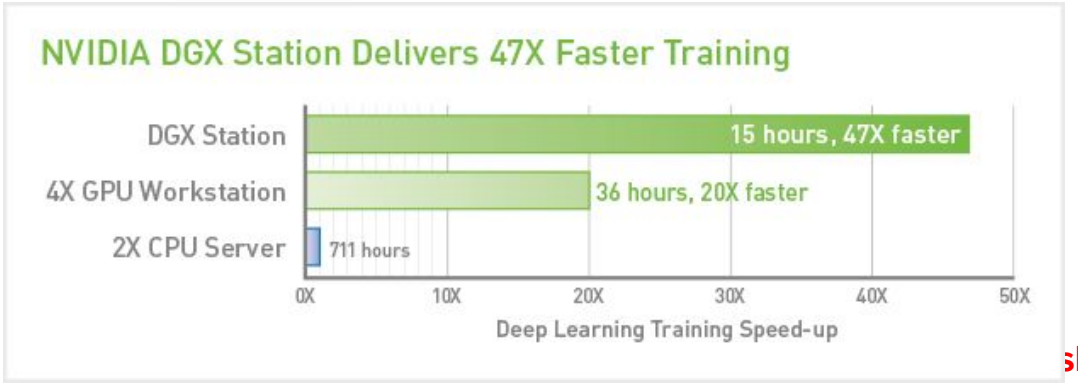
GPGPU – a serviço do aprendizado

- NVIDIA – GeForce Titan X – 2015 – 12GB
- 3072 processadores de Stream, 96 renderizadores
- 192 texturizadores, 24 tesseladores, 6 rasterizadores

Link do slido de hoje: https://app.sli.do/event/bldcpzub

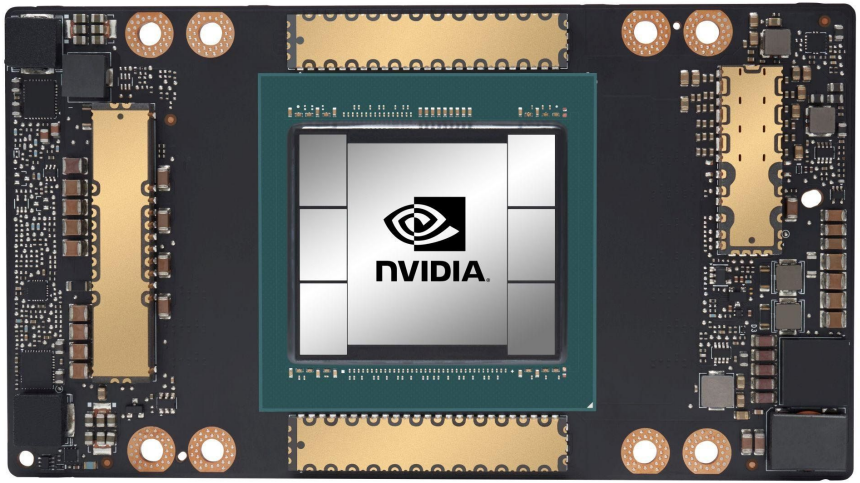
NVIDIA DGX – “personal” AI unit

- Quatro Tesla V100
- 2560 Tensor cores, 20480 CUDA cores
- 480 Tflops (top 500 supercomputers)
- 1.5KW/h (500KW/h)



sli.do/event/bldcpzub

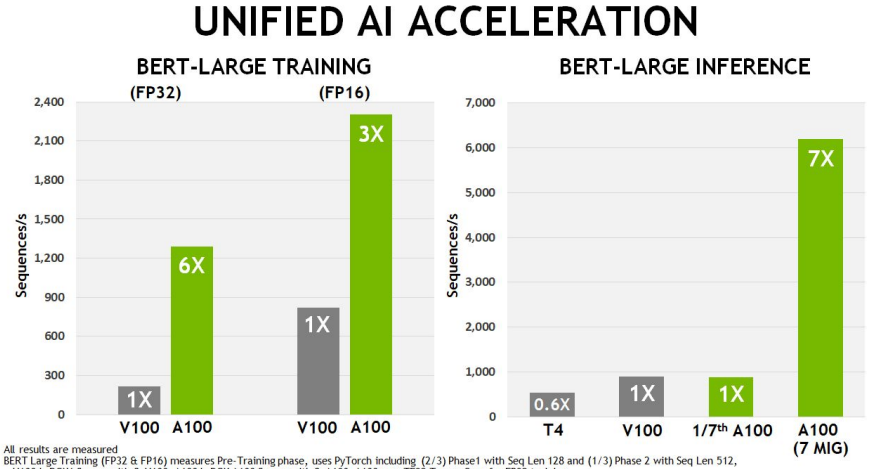
NVIDIA A100 GPU



- 54 bilhões de transistores!!!

  8 GPCs, 8 TPCs/GPC, 2 SMs/TPC, 16 SMs/GPC, 128 SMs per full GPU
  64 FP32 CUDA Cores/SM, 8192 FP32 CUDA Cores per full GPU
  4 Third-generation Tensor Cores/SM, 512 Third-generation Tensor Cores per full GPU
  6 HBM2 stacks, 12 512-bit Memory Controllers

Fonte: nvidia-ampere-architecture-whitepaper.pdf



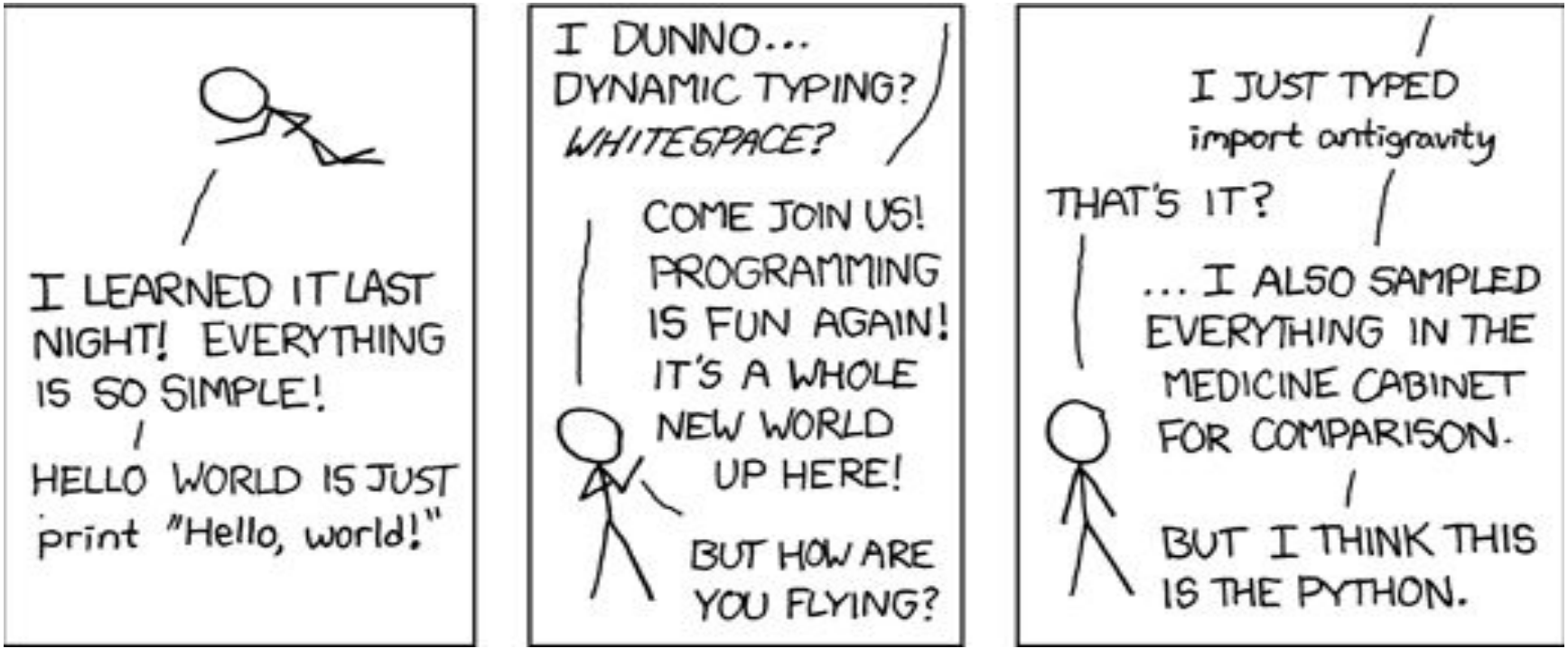
Link do slido de hoje: https://app.sli.do/event/biacpzub

Python

Fonte: XKCD https://xkcd.com/353/

Link do slido de hoje: https://app.sli.do/event/bnd8zpzub

Python



Fonte: XKCD https://xkcd.com/353/

Link do slido de hoje: https://app.sli.do/event/bnd8qpzub

Python

```
net = {}
net['data']= lasagne.layers.InputLayer(data_size,input_var=input_var)
net['conv1'] =
    lasagne.layers.Conv2DLayer(net['data'], num_filters=6,
     filter_size=5)
net['pool1'] = lasagne.layers.Pool2DLayer(net['conv1'], pool_size=2)
net['conv2'] = lasagne.layers.Conv2DLayer(net['pool1'],
     num_filters=10, filter_size=5)
net['pool2'] = lasagne.layers.Pool2DLayer(net['conv2'], pool_size=2)
net['fc1'] = lasagne.layers.DenseLayer(net['pool2'], num_units=100)
net['drop1'] = lasagne.layers.DropoutLayer(net['fc1'],  p=0.5)
net['out'] = lasagne.layers.DenseLayer(net['drop1'],
     num_units=output_size,
     nonlinearity=lasagne.nonlinearities.softmax)
```

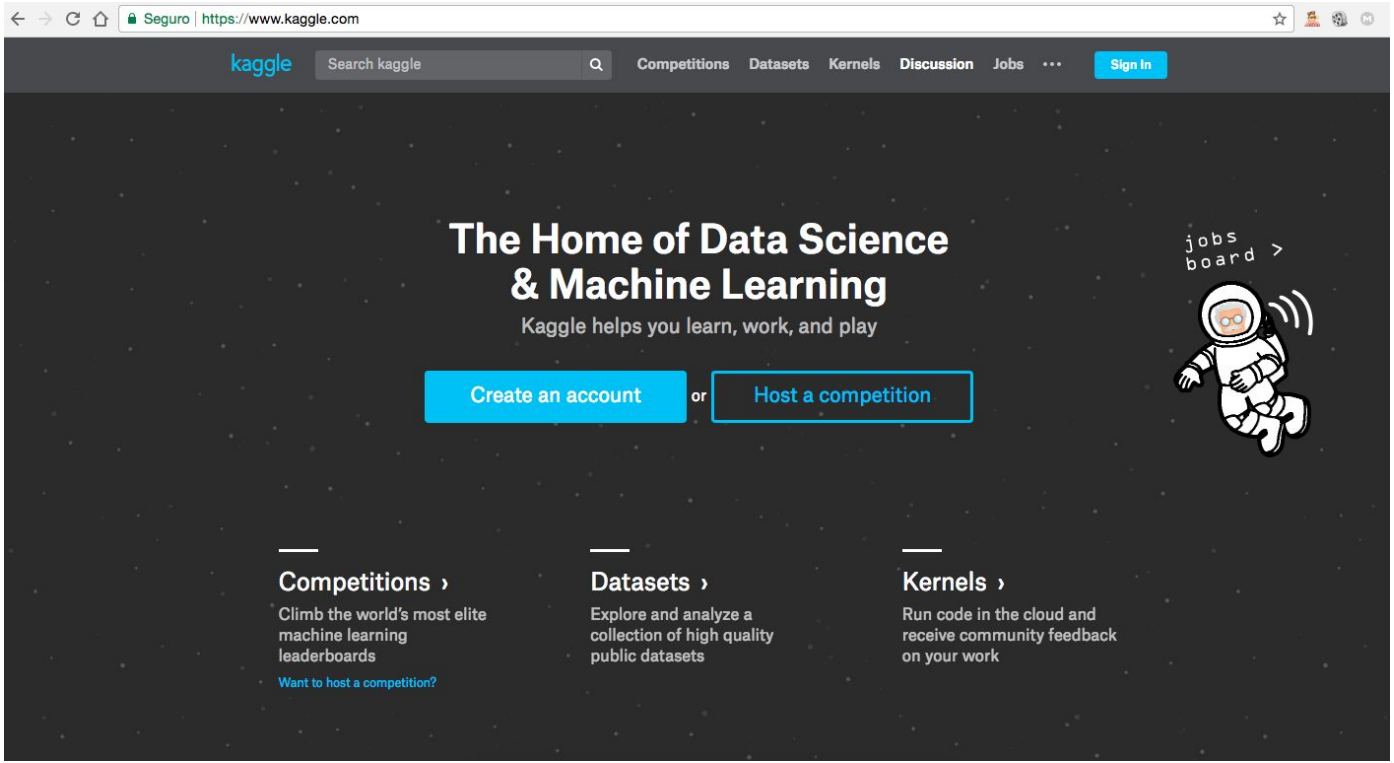
Link do slido de hoje: https://app.sli.do/event/bldcpzub

Competições

- NEURIPS
- CVPR
- ICDAR
- ICIP
- Hackatons de ML
- Kaggle

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Kaggle



Fonte: http://www.kaggle.com

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Kaggle

- World Cup 2010 - Confidence Challenge – US$ 100
- Chess ratings – US$617
- RTA Freeway Travel Time Prediction – US$10000
- Mapping Dark Matter – US$ 3000
- Wikipedia's Participation Challenge – US$ 10000
- Predicting a Biological Response – US$ 20000
- The Hewlett Foundation: Short Answer Scoring – US$ 100000

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Kaggle

- GE Hospital Quest – US$ 100000
- GE Flight Quest - US$ 250000
- Heritage Health Prize – US$ 500000
- The Marinexplore and Cornell University Whale Detection Challenge – US$ 10000
- MasterCard - Data Cleansing Competition – US$ 100000
- AMS 2013-2014 Solar Energy Prediction Contest - 1000
- Galaxy Zoo - The Galaxy Challenge – US$ 16000
- Africa Soil Property Prediction Challenge – US$ 8000

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Kaggle

- National Data Science Bowl – US$ 175000
- Diabetic Retinopathy Detection – US$ 100000
- Facebook V: Predicting Check Ins – Jobs
- Planet: Understanding the Amazon from Space – US$ 60000
- Passenger Screening Algorithm Challenge – US$ 1500000
- Zillow Prize: Zillow's Home Value Prediction (Zestimate) - US$ 1200000

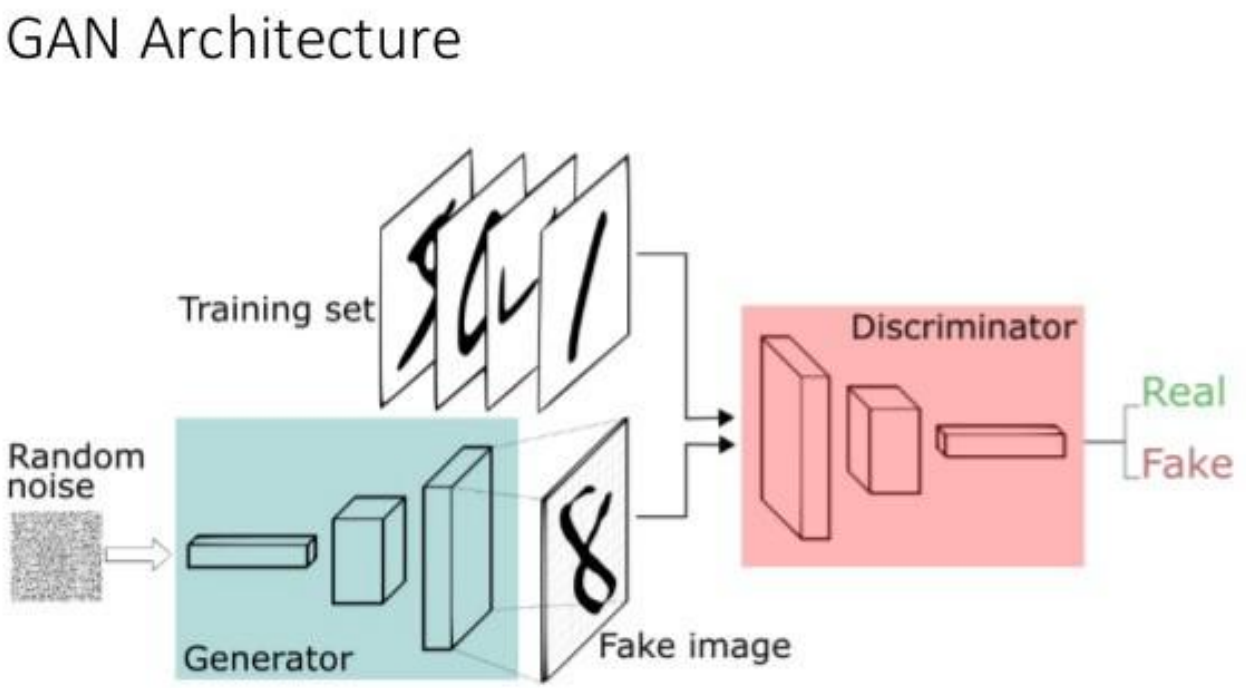
Link do slido de hoje: https://app.sli.do/event/bldcpzub

Desafios recentes

- Não ter dados rotulados suficientes
  - Aumentação de dados
    - De certa forma as Redes Generativas Adversariais (GAN) resolvem parte do problema.
  - Transferência de aprendizado
- Melhor entendimento do que está sendo aprendido
  - visualização das fases intermediárias
  - decomposição do operador (CNN) no domínio das frequências
- Aprendizado de arquiteturas
  - AutoML

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Generative Adversarial Networks



Fonte: https://mc.ai/learning-generative-adversarial-networks-gans/

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Style GAN

Nenhuma destas pessoas existem!!!

Fonte: https://github.com/NVlabs/stylegan

Link do slido de hoje: https://app.sli.do/event/bldcpzub

Desafios recentes e futuros

- Não ter dados rotulados suficientes
  - ainda há muito que pesquisar
- Arquiteturas menores
  - MobileNet etc
- Melhor entendimento do está sendo aprendido?
  - Teoria matemática
  - Capacidade dos modelos
  - Arquiteturas a partir de modelos
  - Como trabalhar com dados imprecisos/com incerteza?

Link do slido de hoje: https://app.sli.do/event/bldcpzub