

# Redes Neurais e Deep Learning

## INTRODUÇÃO À VISÃO COMPUTACIONAL

---

Zenilton K. G. Patrocínio Jr

[zenilton@pucminas.br](mailto:zenilton@pucminas.br)

# Visão Computacional

David Marr: abordagem precursora (1970-80s)



# Visão Computacional

David Marr: abordagem precursora (1970-80s)



Visão  Sistema de processamento de informação

# Visão Computacional

David Marr: abordagem precursora (1970-80s)



Visão → Sistema de processamento de informação

Níveis de análise:

- **Computacional**: o que o sistema faz
- **Algorítmico** : como o sistema faz
- **Físico** : a forma como o sistema é realizado fisicamente

# Visão Computacional

David Marr: abordagem precursora (1970-80s)



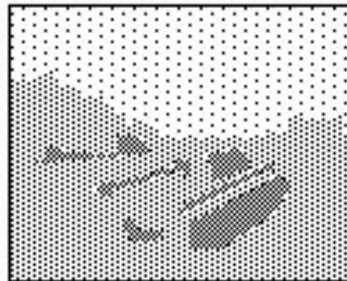
Visão  Sistema de processamento de informação

Níveis de análise:

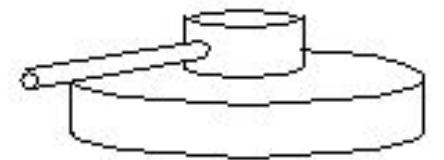
- **Computacional**: o que o sistema faz
- **Algorítmico** : como o sistema faz
- **Físico** : a forma como o sistema é realizado fisicamente

**Estágios  
da Visão**

input image



3-D model



# Visão Computacional

David Marr: abordagem precursora (1970-80s)



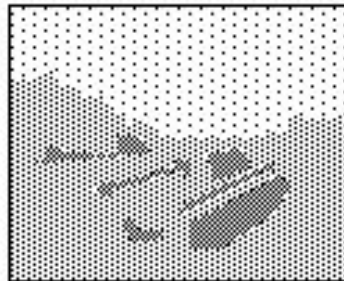
Visão  Sistema de processamento de informação

Níveis de análise:

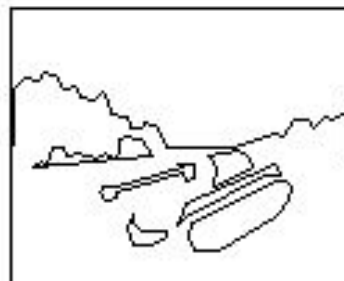
- **Computacional**: o que o sistema faz
- **Algorítmico** : como o sistema faz
- **Físico** : a forma como o sistema é realizado fisicamente

**Estágios  
da Visão**

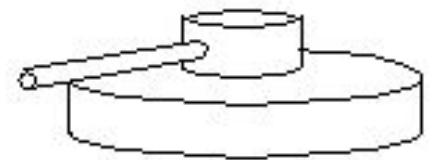
input image



edge image



3-D model



# Visão Computacional

David Marr: abordagem precursora (1970-80s)



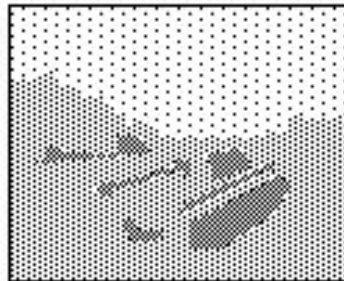
Visão  Sistema de processamento de informação

Níveis de análise:

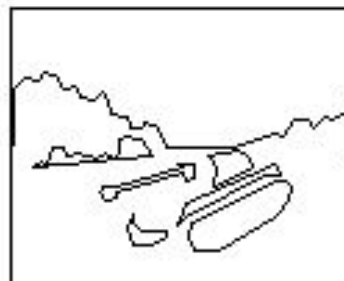
- **Computacional**: o que o sistema faz
- **Algorítmico** : como o sistema faz
- **Físico** : a forma como o sistema é realizado fisicamente

**Estágios  
da Visão**

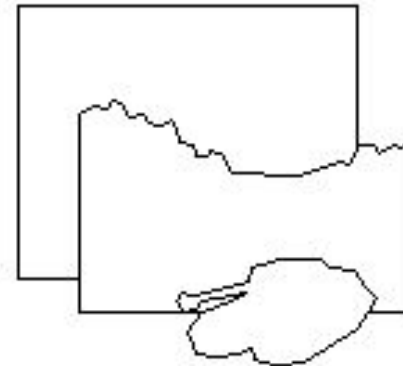
input image



edge image



2<sup>1</sup>/<sub>2</sub>-D sketch



3-D model

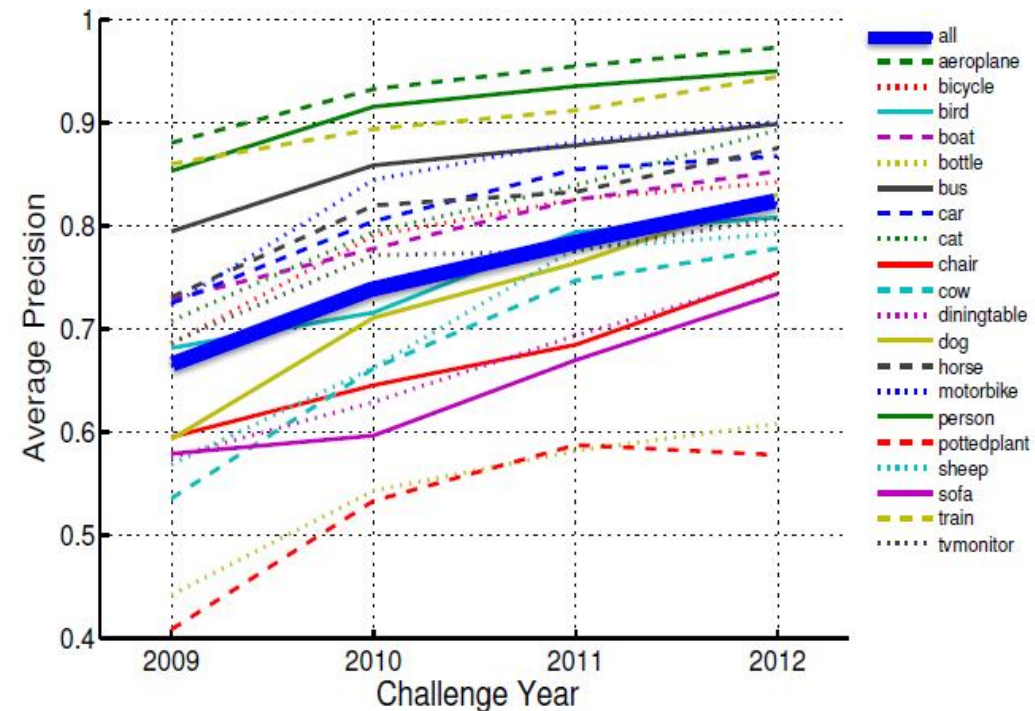




# Visão Computacional – *Datasets Desafiadores*

## PASCAL Visual Object Challenge (20 object categories)

[Everingham et al. 2006-2012]

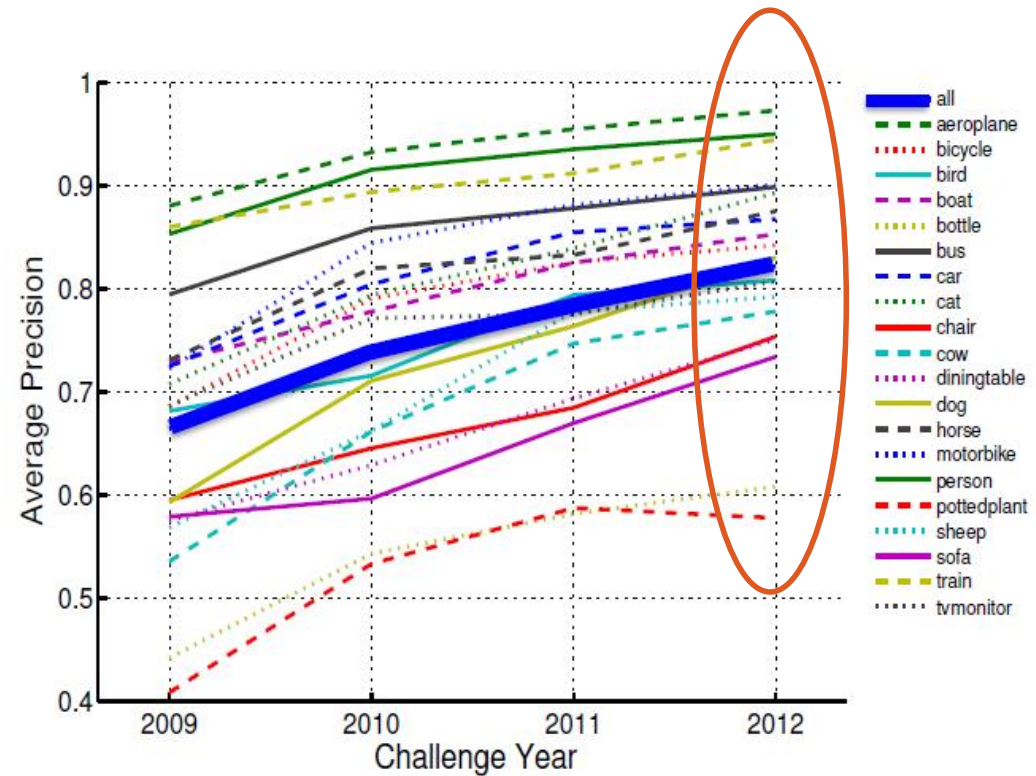




# Visão Computacional – *Datasets* Desafiadores

## PASCAL Visual Object Challenge (20 object categories)

[Everingham et al. 2006-2012]






# Visão Computacional – *Datasets* Desafiadores



IMGENET

[www.image-net.org](http://www.image-net.org)

 **22K** categories and **14M** images

- Animals
  - Bird
  - Fish
  - Mammal
  - Invertebrate
- Plants
  - Tree
  - Flower
  - Food
  - Materials
- Structures
  - Artifact
  - Tools
  - Appliances
  - Structures
- Person
  - Scenes
    - Indoor
    - Geological Formations
  - Sport Activities



Deng, Dong, Socher, Li, Li, & Fei-Fei, 2009



# Visão Computacional – *Datasets* Desafiadores

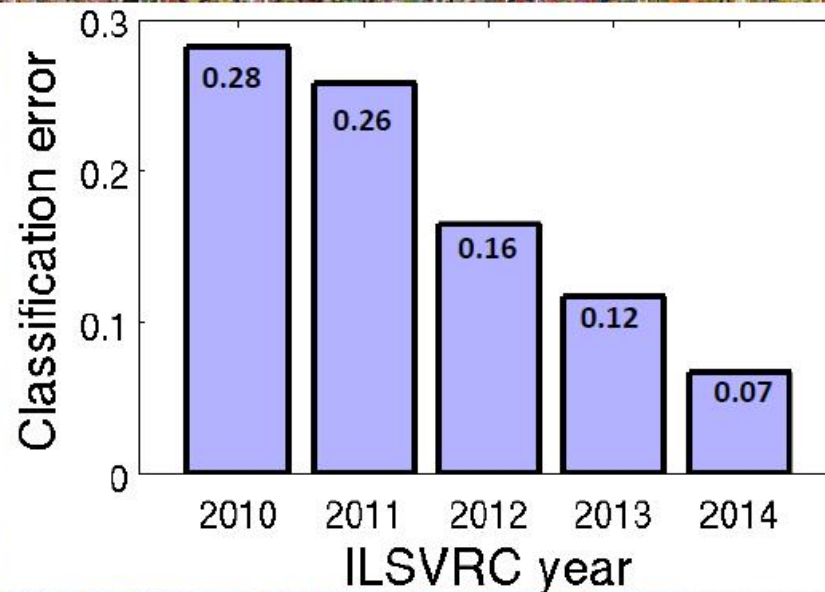
## IMAGENET Large Scale Visual Recognition Challenge

Steel drum

The Image Classification Challenge:

1,000 object classes

1,431,167 images



Russakovsky et al. arXiv, 2014

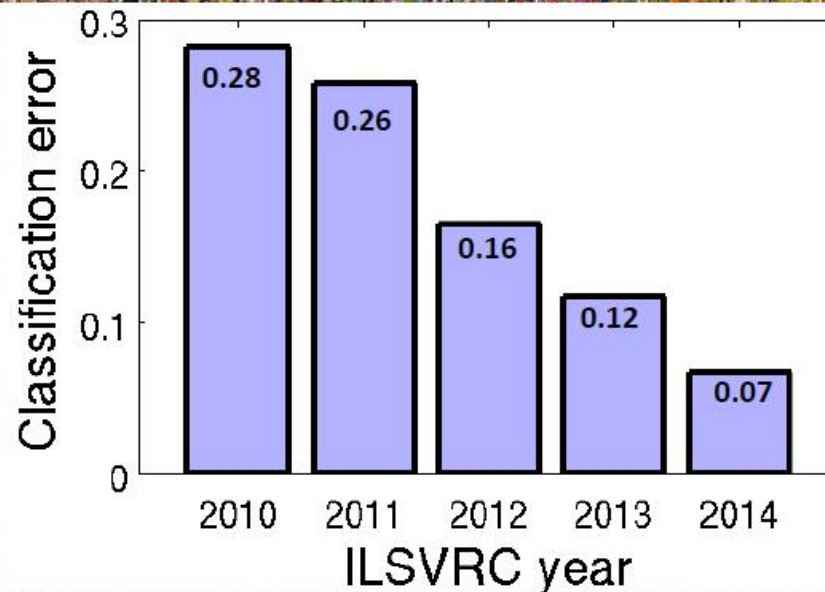


# Visão Computacional – *Datasets* Desafiadores

## IMAGENET Large Scale Visual Recognition Challenge

The Image Classification Challenge:

1,000 object classes  
1,431,167 images



Russakovsky et al. arXiv, 2014

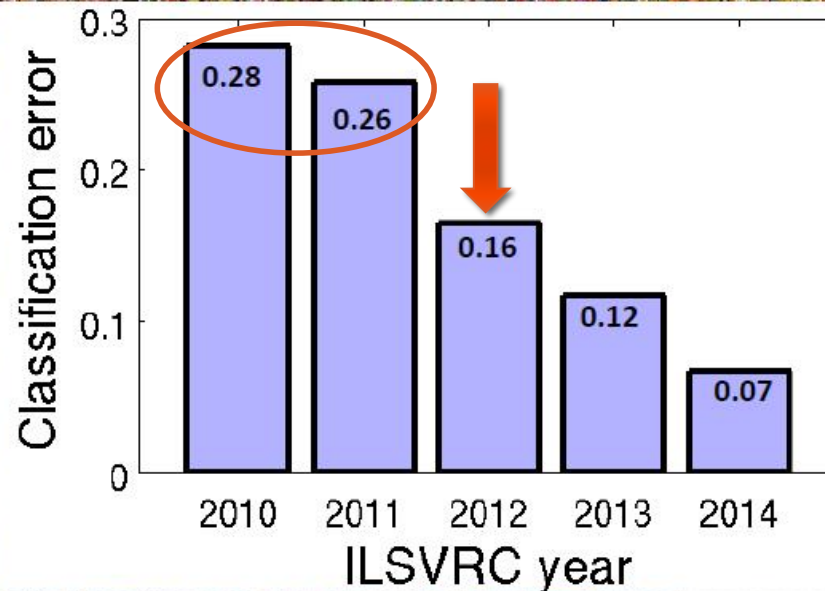


# Visão Computacional – *Datasets* Desafiadores

## IMAGENET Large Scale Visual Recognition Challenge

The Image Classification Challenge:

1,000 object classes  
1,431,167 images



Russakovsky et al. arXiv, 2014

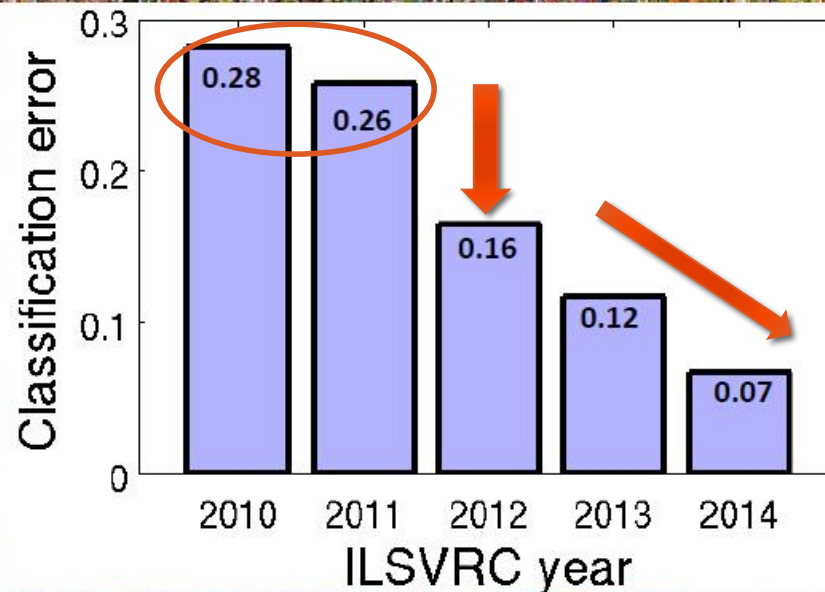


# Visão Computacional – *Datasets* Desafiadores

## IMAGENET Large Scale Visual Recognition Challenge

The Image Classification Challenge:

1,000 object classes  
1,431,167 images



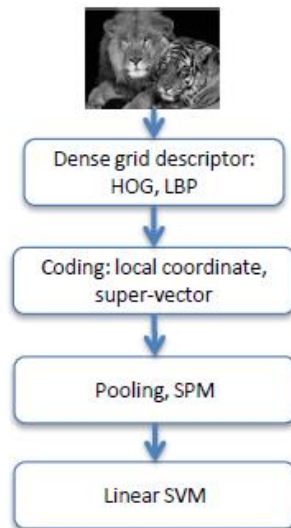
Russakovsky et al. arXiv, 2014

# Visão Computacional – Aprendizagem Profunda

## IMAGENET Large Scale Visual Recognition Challenge

Year 2010

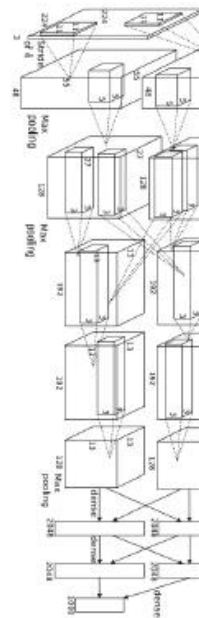
NEC-UIUC



[Lin CVPR 2011]

Year 2012

SuperVision



[Krizhevsky NIPS 2012]

Year 2014

GoogLeNet



[Szegedy arxiv 2014]

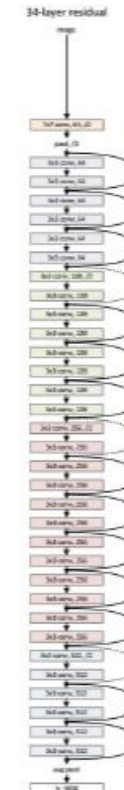
VGG



[Simonyan arxiv 2014]

Year 2015

MSRA





# Classificação de Imagem – Tarefa Fundamental

Dada uma imagem, como  
por exemplo:



# Classificação de Imagem – Tarefa Fundamental

Dada uma imagem, como por exemplo:



E assumindo um conjunto de rótulos (ou classes) discretos

Por exemplo: cachorro, gato, caminhão, avião, ...

# Classificação de Imagem – Tarefa Fundamental

Dada uma imagem, como por exemplo:



E assumindo um conjunto de rótulos (ou classes) discretos

Por exemplo: cachorro, gato, caminhão, avião, ...

Classificar

gato

# Problema: “*Gap*” Semântico

Imagens são representadas  
como matrizes 3D de números,  
geralmente, inteiros entre 0 e  
255



# Problema: “*Gap*” Semântico

Imagens são representadas como matrizes 3D de números, geralmente, inteiros entre 0 e 255

Por exemplo, para uma imagem de dimensões 300 por 100 em 3 canais de cor temos:

$300 \times 100 \times 3 = 90.000$  números



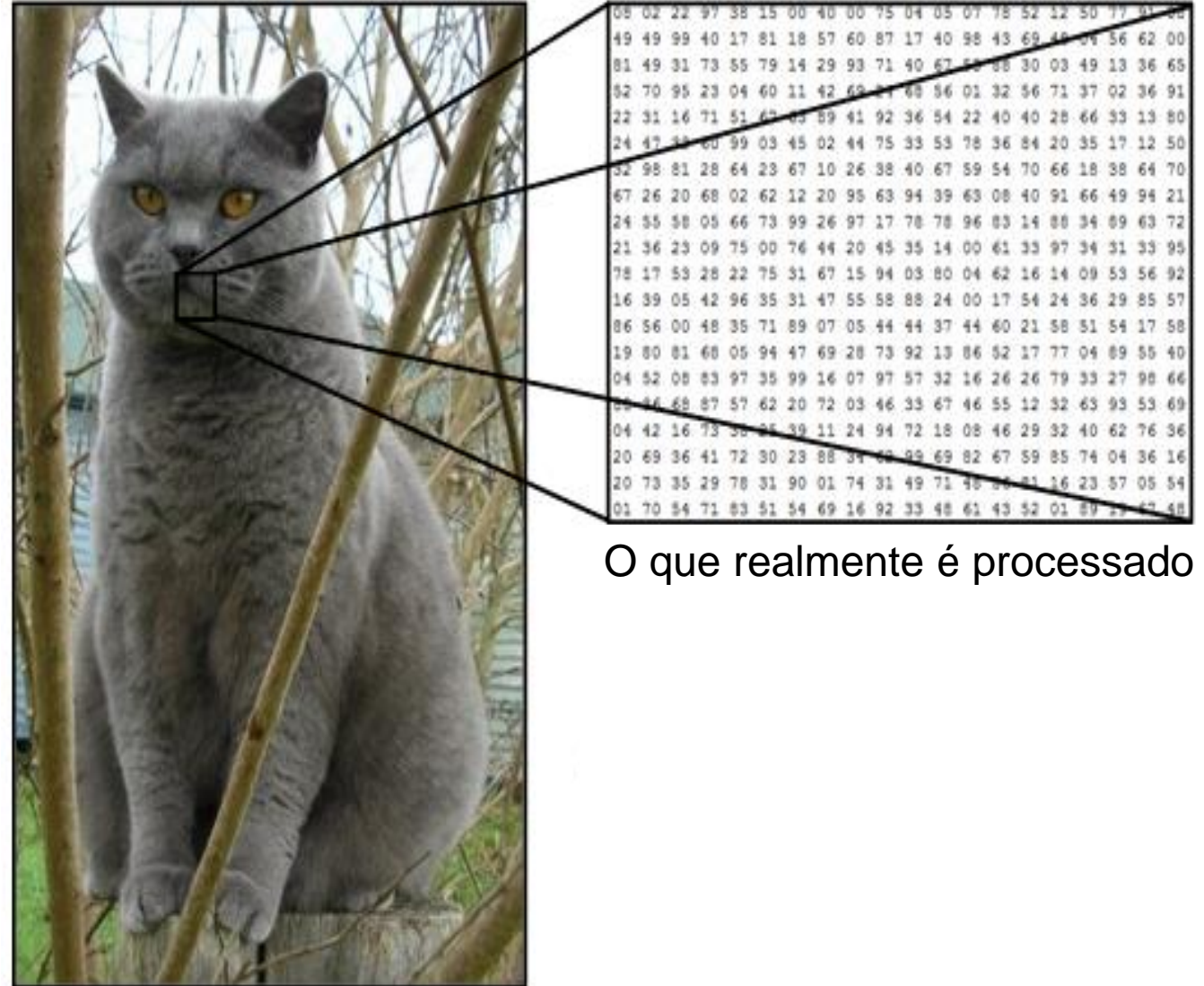


# Problema: “Gap” Semântico

Imagens são representadas como matrizes 3D de números, geralmente, inteiros entre 0 e 255

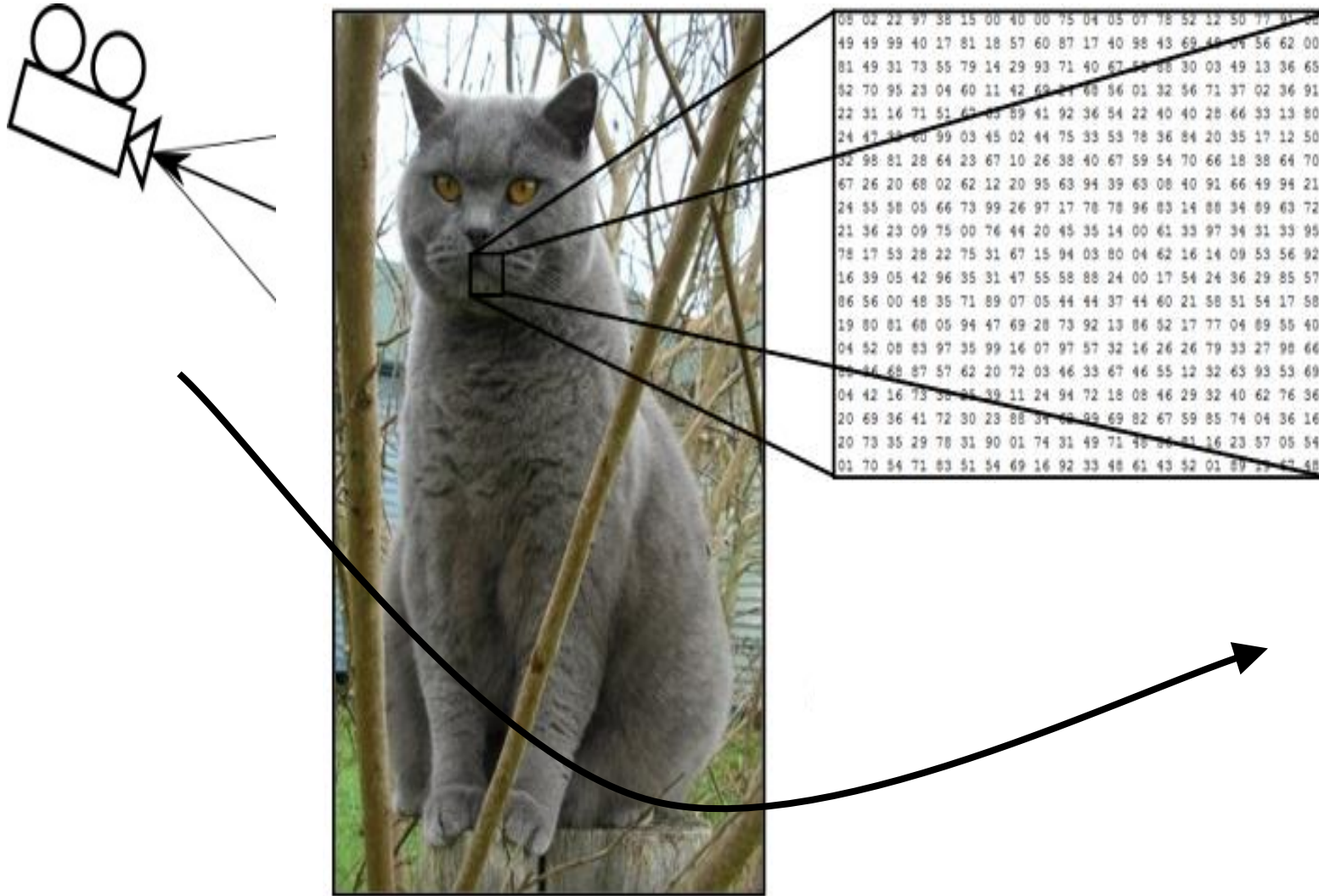
Por exemplo, para uma imagem de dimensões 300 por 100 em 3 canais de cor temos:

$300 \times 100 \times 3 = 90.000$  números



O que realmente é processado

# Desafios: Variação de Ponto de Vista

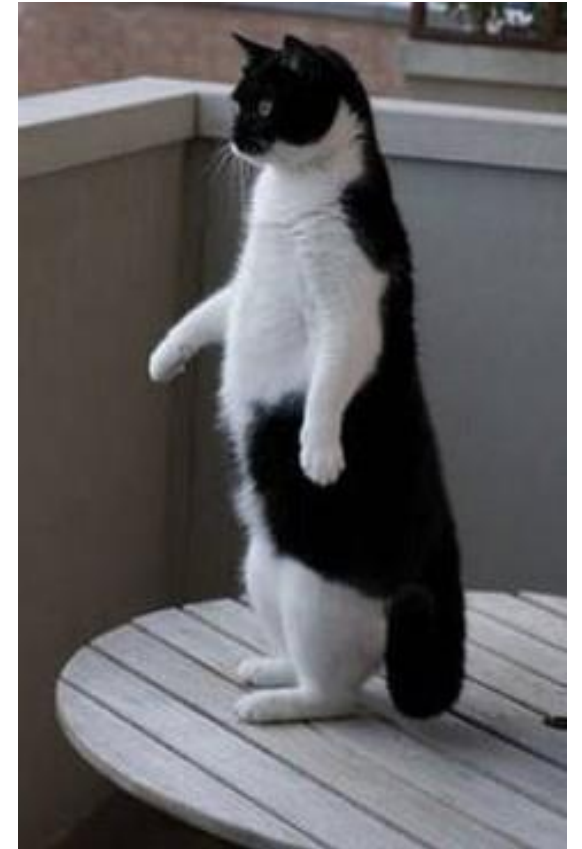




# Desafios: Variação de Iluminação



# Desafios: Deformação



# Desafios: Oclusão





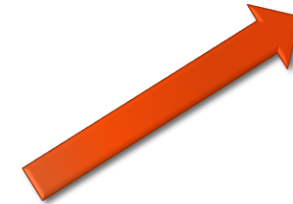
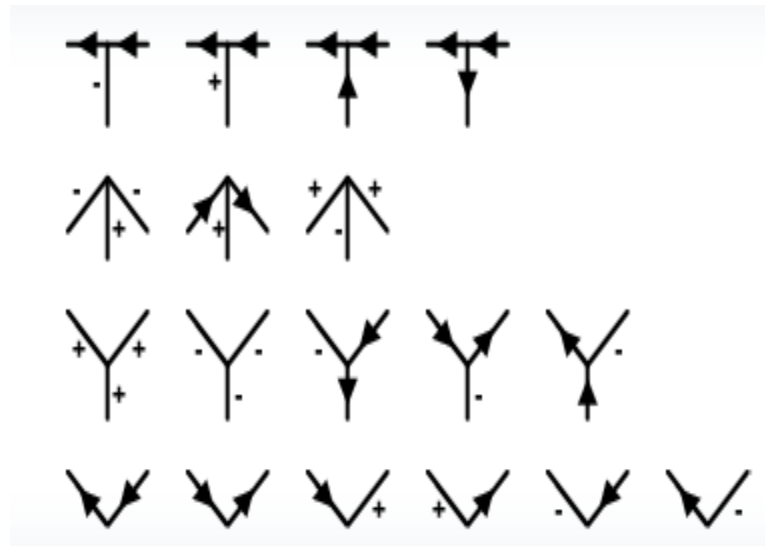
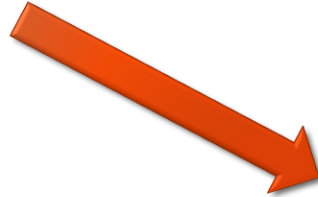
# Desafios: Separação do Fundo



# Desafios: Variação Intraclasse



# Classificação usando Regras ou Restrições



?



# Abordagem Baseada em Dados

## 1. Reune-se um conjunto de imagens com seus rótulos

### Exemplo de conjunto de treinamento

gato

cão

caneca

chapéu





# Abordagem Baseada em Dados

1. Reune-se um conjunto de imagens com seus rótulos
2. Usa-se de método de Aprendizado de Máquina para treinar um classificador

## Exemplo de conjunto de treinamento

gato



cão



caneca



chapéu



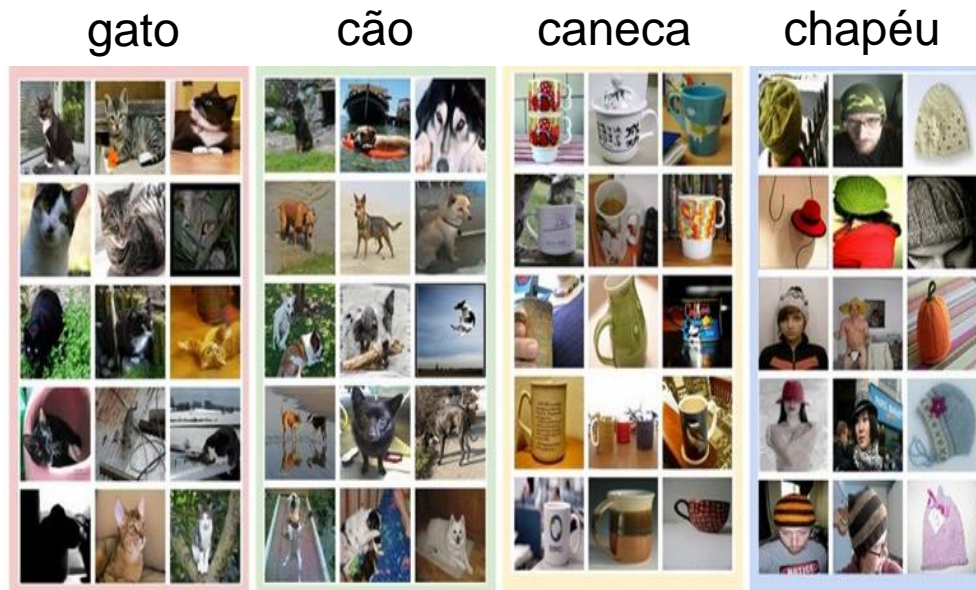
## Classificador

```
def train(train_images, train_labels):  
    # build a model for images -> labels...  
    return model  
  
def predict(model, test_images):  
    # predict test_labels using the model...  
    return test_labels
```

# Abordagem Baseada em Dados

1. Reune-se um conjunto de imagens com seus rótulos
2. Usa-se de método de Aprendizado de Máquina para treinar um classificador
3. Avalia-se o classificador com um conjunto separado de imagens de teste

## Exemplo de conjunto de treinamento



## Classificador

```
def train(train_images, train_labels):  
    # build a model for images -> labels...  
    return model  
  
def predict(model, test_images):  
    # predict test_labels using the model...  
    return test_labels
```

# Classificador Simples: Vizinho Mais Próximo

## Classificador


```
def train(train_images, train_labels):  
    # build a model for images -> labels...  
    return model  
  
def predict(model, test_images):  
    # predict test_labels using the model...  
    return test_labels
```

# Classificador Simples: Vizinho Mais Próximo

## Classificador

```
def train(train_images, train_labels):  
    # build a model for images -> labels...  
    return model  
  
def predict(model, test_images):  
    # predict test_labels using the model...  
    return test_labels
```

Lembrar de todas as imagens  
de treinamento e seus rótulos



# Classificador Simples: Vizinho Mais Próximo

## Classificador

```
def train(train_images, train_labels):  
    # build a model for images -> labels...  
    return model  
  
def predict(model, test_images):  
    # predict test_labels using the model...  
    return test_labels
```

Lembrar de todas as imagens de treinamento e seus rótulos

Predizer o rótulo a partir da imagem de treinamento mais similar

# Classificador Simples: Vizinho Mais Próximo

Como avaliar a similaridade entre imagens?



# Classificador Simples: Vizinho Mais Próximo

Como avaliar a similaridade entre imagens?  Uso de distância



# Classificador Simples: Vizinho Mais Próximo

Como avaliar a similaridade entre imagens?  Uso de distância

**Distância L1:**  $d_1(I_1, I_2) = \sum_p |I_1^p - I_2^p|$

**Distância L2:**  $d_2(I_1, I_2) = \sqrt{\sum_p (I_1^p - I_2^p)^2}$

# Classificador Simples: Vizinho Mais Próximo

Como avaliar a similaridade entre imagens?  Uso de distância

**Distância L1:**  $d_1(I_1, I_2) = \sum_p |I_1^p - I_2^p|$

**Distância L2:**  $d_2(I_1, I_2) = \sqrt{\sum_p (I_1^p - I_2^p)^2}$

## Exemplo com Distância L1

Imagem de teste

56	32	10	18
90	23	128	133
24	26	178	200
2	0	255	220

# Classificador Simples: Vizinho Mais Próximo

Como avaliar a similaridade entre imagens?  Uso de distância

**Distância L1:**  $d_1(I_1, I_2) = \sum_p |I_1^p - I_2^p|$

**Distância L2:**  $d_2(I_1, I_2) = \sqrt{\sum_p (I_1^p - I_2^p)^2}$

## Exemplo com Distância L1

Imagem de teste

56	32	10	18
90	23	128	133
24	26	178	200
2	0	255	220

Imagem de treino

10	20	24	17
8	10	89	100
12	16	178	170
4	32	233	112

# Classificador Simples: Vizinho Mais Próximo

Como avaliar a similaridade entre imagens?  Uso de distância

**Distância L1:**  $d_1(I_1, I_2) = \sum_p |I_1^p - I_2^p|$

**Distância L2:**  $d_2(I_1, I_2) = \sqrt{\sum_p (I_1^p - I_2^p)^2}$

## Exemplo com Distância L1

Imagem de teste

56	32	10	18
90	23	128	133
24	26	178	200
2	0	255	220

-

Imagem de treino

10	20	24	17
8	10	89	100
12	16	178	170
4	32	233	112

# Classificador Simples: Vizinho Mais Próximo

Como avaliar a similaridade entre imagens?  Uso de distância

**Distância L1:**  $d_1(I_1, I_2) = \sum_p |I_1^p - I_2^p|$

**Distância L2:**  $d_2(I_1, I_2) = \sqrt{\sum_p (I_1^p - I_2^p)^2}$

## Exemplo com Distância L1

Imagem de teste		Imagem de treino		Valor abs. diferenças por pixel																																																
<table><tr><td>56</td><td>32</td><td>10</td><td>18</td></tr><tr><td>90</td><td>23</td><td>128</td><td>133</td></tr><tr><td>24</td><td>26</td><td>178</td><td>200</td></tr><tr><td>2</td><td>0</td><td>255</td><td>220</td></tr></table>	56	32	10	18	90	23	128	133	24	26	178	200	2	0	255	220	-	<table><tr><td>10</td><td>20</td><td>24</td><td>17</td></tr><tr><td>8</td><td>10</td><td>89</td><td>100</td></tr><tr><td>12</td><td>16</td><td>178</td><td>170</td></tr><tr><td>4</td><td>32</td><td>233</td><td>112</td></tr></table>	10	20	24	17	8	10	89	100	12	16	178	170	4	32	233	112	=	<table><tr><td>46</td><td>12</td><td>14</td><td>1</td></tr><tr><td>82</td><td>13</td><td>39</td><td>33</td></tr><tr><td>12</td><td>10</td><td>0</td><td>30</td></tr><tr><td>2</td><td>32</td><td>22</td><td>108</td></tr></table>	46	12	14	1	82	13	39	33	12	10	0	30	2	32	22	108
56	32	10	18																																																	
90	23	128	133																																																	
24	26	178	200																																																	
2	0	255	220																																																	
10	20	24	17																																																	
8	10	89	100																																																	
12	16	178	170																																																	
4	32	233	112																																																	
46	12	14	1																																																	
82	13	39	33																																																	
12	10	0	30																																																	
2	32	22	108																																																	



# Classificador Simples: Vizinho Mais Próximo

Como avaliar a similaridade entre imagens?  Uso de distância

**Distância L1:**  $d_1(I_1, I_2) = \sum_p |I_1^p - I_2^p|$

**Distância L2:**  $d_2(I_1, I_2) = \sqrt{\sum_p (I_1^p - I_2^p)^2}$

## Exemplo com Distância L1

Imagem de teste		Imagem de treino		Valor abs. diferenças por pixel																																																	
<table><tr><td>56</td><td>32</td><td>10</td><td>18</td></tr><tr><td>90</td><td>23</td><td>128</td><td>133</td></tr><tr><td>24</td><td>26</td><td>178</td><td>200</td></tr><tr><td>2</td><td>0</td><td>255</td><td>220</td></tr></table>	56	32	10	18	90	23	128	133	24	26	178	200	2	0	255	220	-	<table><tr><td>10</td><td>20</td><td>24</td><td>17</td></tr><tr><td>8</td><td>10</td><td>89</td><td>100</td></tr><tr><td>12</td><td>16</td><td>178</td><td>170</td></tr><tr><td>4</td><td>32</td><td>233</td><td>112</td></tr></table>	10	20	24	17	8	10	89	100	12	16	178	170	4	32	233	112	=	<table><tr><td>46</td><td>12</td><td>14</td><td>1</td></tr><tr><td>82</td><td>13</td><td>39</td><td>33</td></tr><tr><td>12</td><td>10</td><td>0</td><td>30</td></tr><tr><td>2</td><td>32</td><td>22</td><td>108</td></tr></table>	46	12	14	1	82	13	39	33	12	10	0	30	2	32	22	108	$\Sigma \rightarrow 456$
56	32	10	18																																																		
90	23	128	133																																																		
24	26	178	200																																																		
2	0	255	220																																																		
10	20	24	17																																																		
8	10	89	100																																																		
12	16	178	170																																																		
4	32	233	112																																																		
46	12	14	1																																																		
82	13	39	33																																																		
12	10	0	30																																																		
2	32	22	108																																																		

# Exemplo – *Dataset* CIFAR-10

**10** classes

**50.000** imagens para treino

**10.000** imagens para teste

avião



carro



pássaro



gato



corça



cão



rã



cavalo



navio



caminhão



# Exemplo – *Dataset* CIFAR-10

**10** classes

**50.000** imagens para treino

**10.000** imagens para teste

avião



carro



pássaro



gato



corça



cão



rã



cavalo



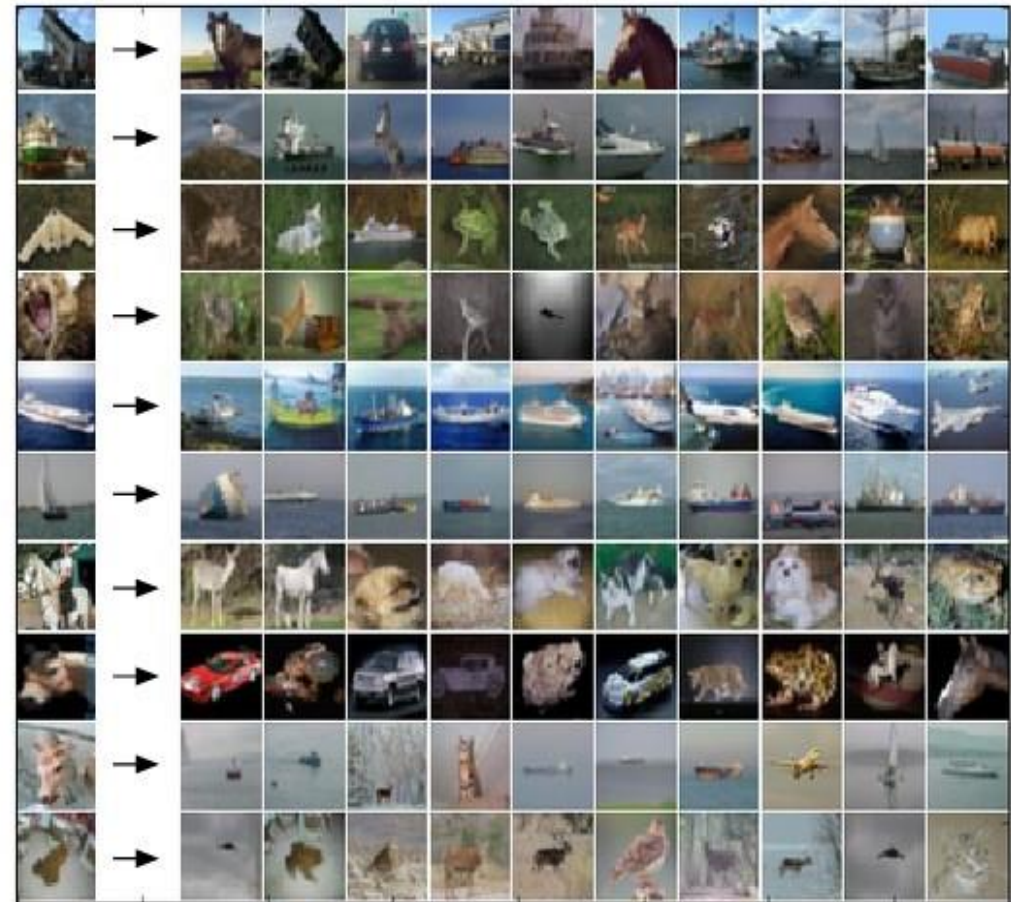
navio



caminhão



Exemplos dos vizinhos mais próximos para cada imagem teste (primeira coluna)



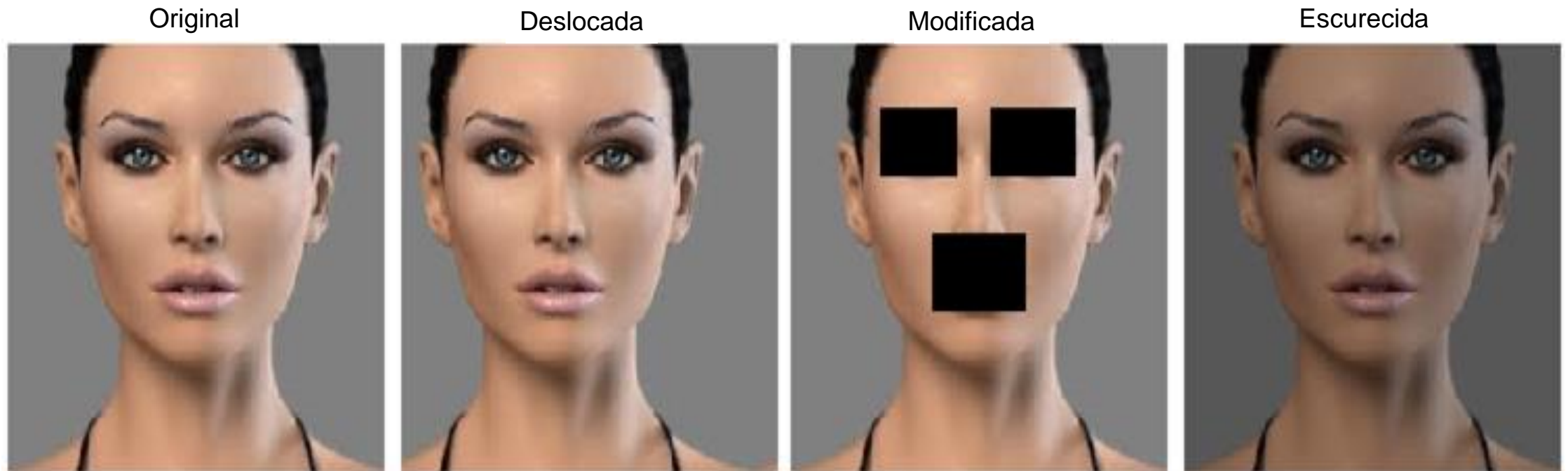


# Classificador Simples: Vizinho Mais Próximo

- Raramente utilizado sobre as imagens “brutas”
- Pode apresentar acurácia ruim durante os testes
- Distâncias entre imagens inteiras podem ser pouco intuitivas

# Classificador Simples: Vizinho Mais Próximo

- Raramente utilizado sobre as imagens “brutas”
- Pode apresentar acurácia ruim durante os testes
- Distâncias entre imagens inteiras podem ser pouco intuitivas



(as 3 imagens mais a direita possuem a mesma distância L2 em relação a primeira)