

# Inteligência Artificial

BCC-35G

Ciência da Computação

Diego Bertolini

[diegobertolini@utfpr.edu.br](mailto:diegobertolini@utfpr.edu.br)

# Extração de Características

- Entender os conceitos de padrões (formas).
- O que são características e por que elas são importantes.
- Introdução a percepção
- Apresentar diferentes tipos de características.

# Padrões

- A facilidade com que nós humanos classificamos e interpretamos os padrões que nos cercam, dão a falsa idéia de que é fácil automatizar tal processo.

**Como reconhecemos um determinado padrão?**

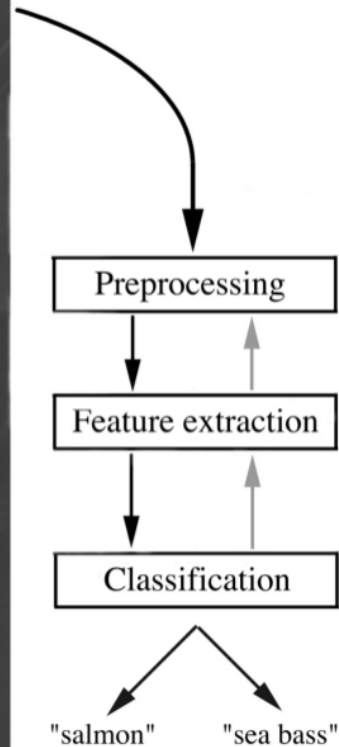
Mesmo sem perceber, extraímos suas características relevantes.

# Características

Qualquer medida que se possa extrair de um determinado objeto.

- Simbólicas
- Numéricas contínuas.
- Numéricas binárias.

# Características



# Características

Exemplo de característica simbólica:

- Cor do objeto.

Exemplo de característica numérica continua.

- Peso do objeto.

Característica numérica binária.

- Determinam a presença ou ausência de uma determinada característica.



# Características

**Objetivo:** Caracterizar um objeto através de medidas, as quais são bastante *similares para objetos da mesma classe*, e bastante *diferentes para objetos de outras classes*.

- **Características discriminantes e invariantes.**

# Características

## **Globais**

Extrair características de uma maneira global usando todo sinal.

Maneira como os humanos reconhecem objetos

Gestalt (Percepção)

## **Locais**

Segmentar em partes menores para então extrair características.



# Percepção

Processo de adquirir, interpretar, selecionar e organização informações sensoriais.

## **Gestalt**

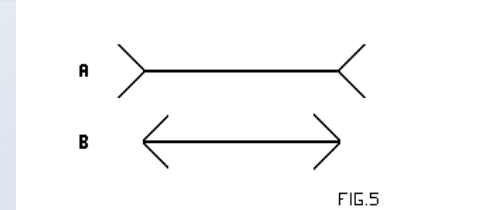
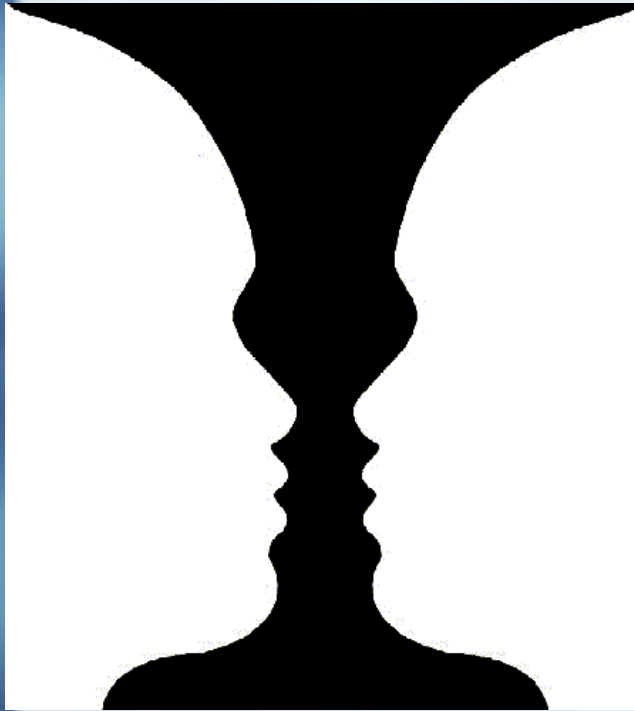
Enfatiza o todo

Processo de reconhecimento se dá pelas propriedades globais (holístico) e não pelas partes.

Baseia-se em princípios tais como

- Emergência, Construtivismo, Invariância

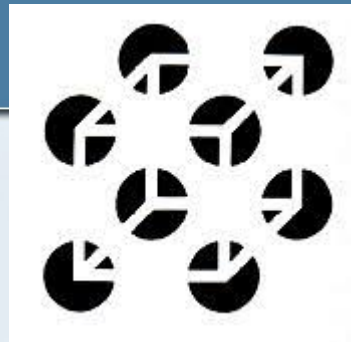
# Gestalt



## Emergência

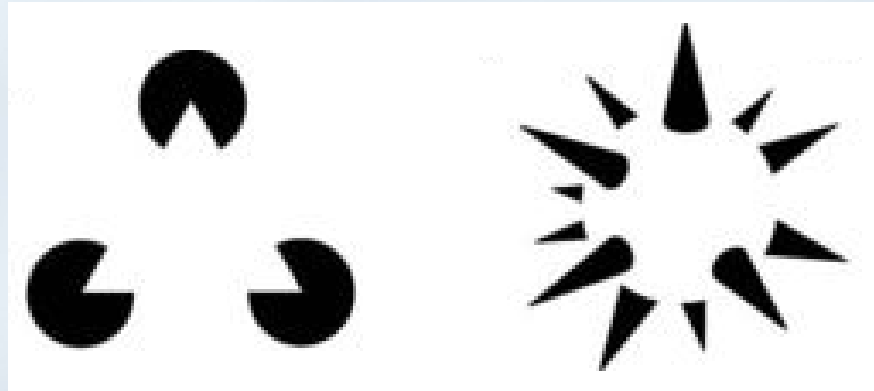
Prova de que reconhecemos a partir do todo e não por partes.

# Gestalt



Prova de que reconhecemos a partir do todo e não por partes.

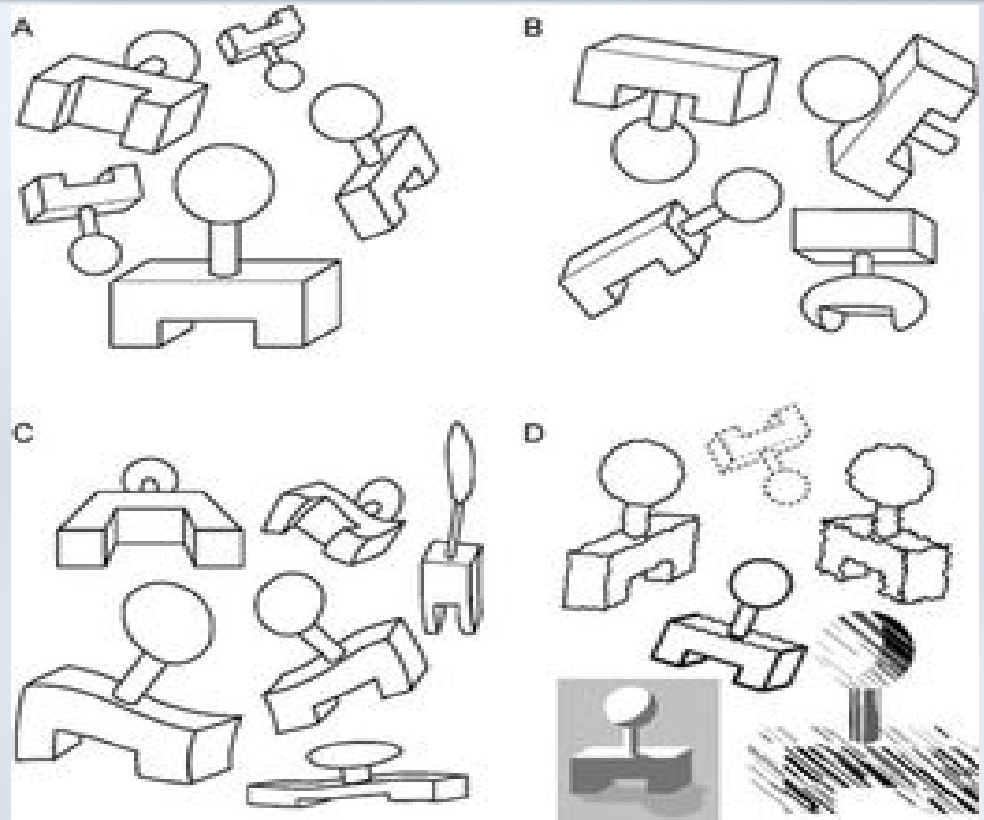
# Gestalt



## Construtivismo

Conseguimos identificar características não presentes a partir da percepção de características presentes.

# Gestalt



## Invariância

Objetos são reconhecidos independentemente de rotação, translação, escala e ruído.

# Holístico X Local

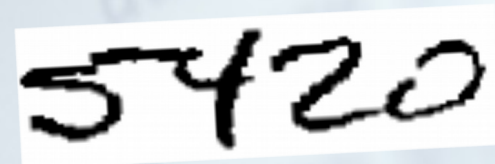
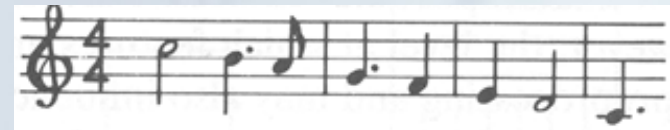
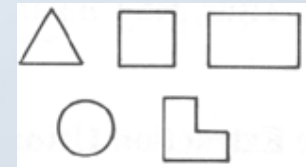
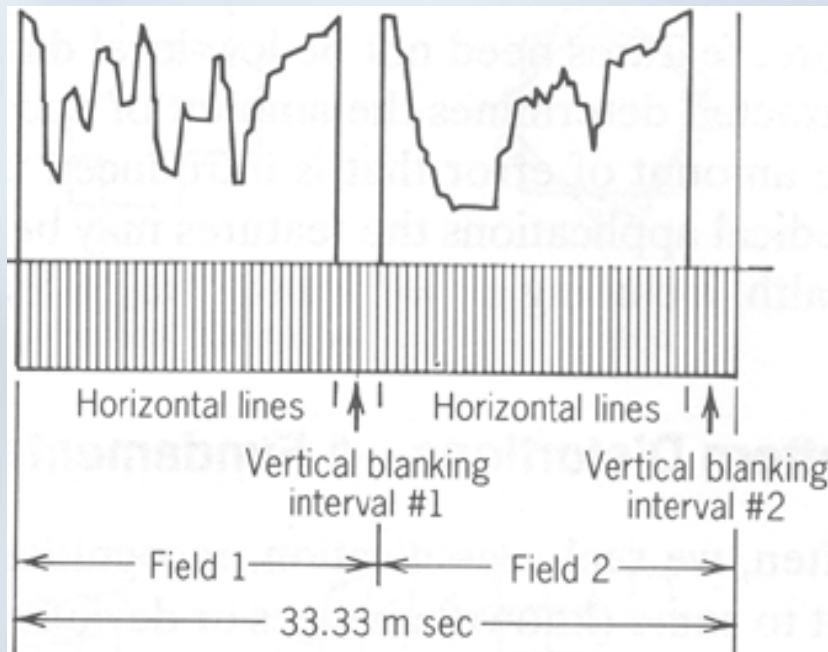
Muitas vezes características globais, como as defendida pela Gestalt, não apresentam desempenho satisfatório.

Nesses casos, características locais se tornam bastante interessantes.



# Padrões

A maioria das coisas que nos cerca podem ser definidas como **padrões**.



# Características

No nosso exemplo dos peixes:  
Devemos procurar características  
invariantes a rotação e translação.  
Não sabemos como o peixe estará  
posicionado na esteira.

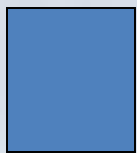
Tamanho é uma boa característica?

- Não, pois um salmão jovem é menor que um salmão adulto, mas continua salmão (**escala**).

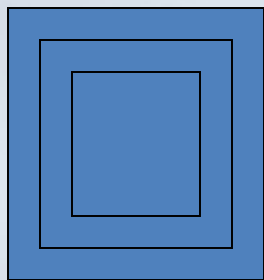
# Características

Então que tipo de características deveríamos empregar?

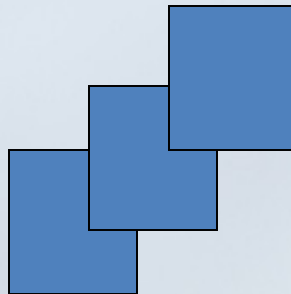
Características ligadas a cor e textura geralmente são invariantes a rotação e translação.



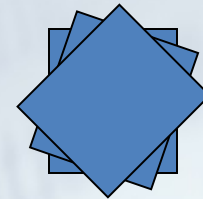
Padrão



Padrão dilatado



Translação



Rotação

# Características Estatísticas

Geralmente extraem coeficientes estatísticos do padrão como um todo. Entre elas podemos citar

*Templates*

Momentos de

- Hu (invariante a rotação, translação e escala)
- Zernike (invariante a rotação)

PCA (*Principal Component Analysis*)

Correlação

# Template Matching

Usa as características de mais baixo nível conhecidas:

- Pixel

O processo é simples e funciona quando os padrões são bem comportados.

Basicamente consiste em:

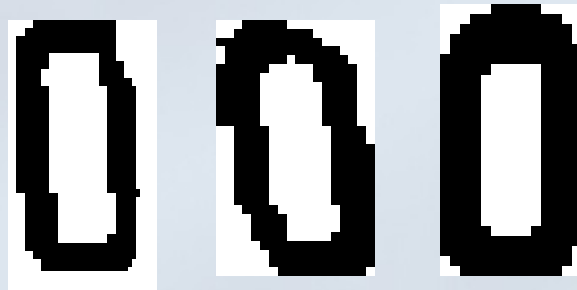
Criar um *template* para cada classe do problema em questão.

Comparar o exemplo de teste com todos os *templates* disponíveis.

# Template Matching

Ruídos devido aquisição

Aumentam a variabilidade, diminuindo assim a eficiência do *template matching*





# Template Matching

Uma outra forma de usar *template matching* consiste em fazer a comparação usando um esquema de zoneamento.

- Enfatizar diferenças locais

Uma variante do *template matching* é o *feature matching*.

Nesse caso, a comparação se dá no nível das características.

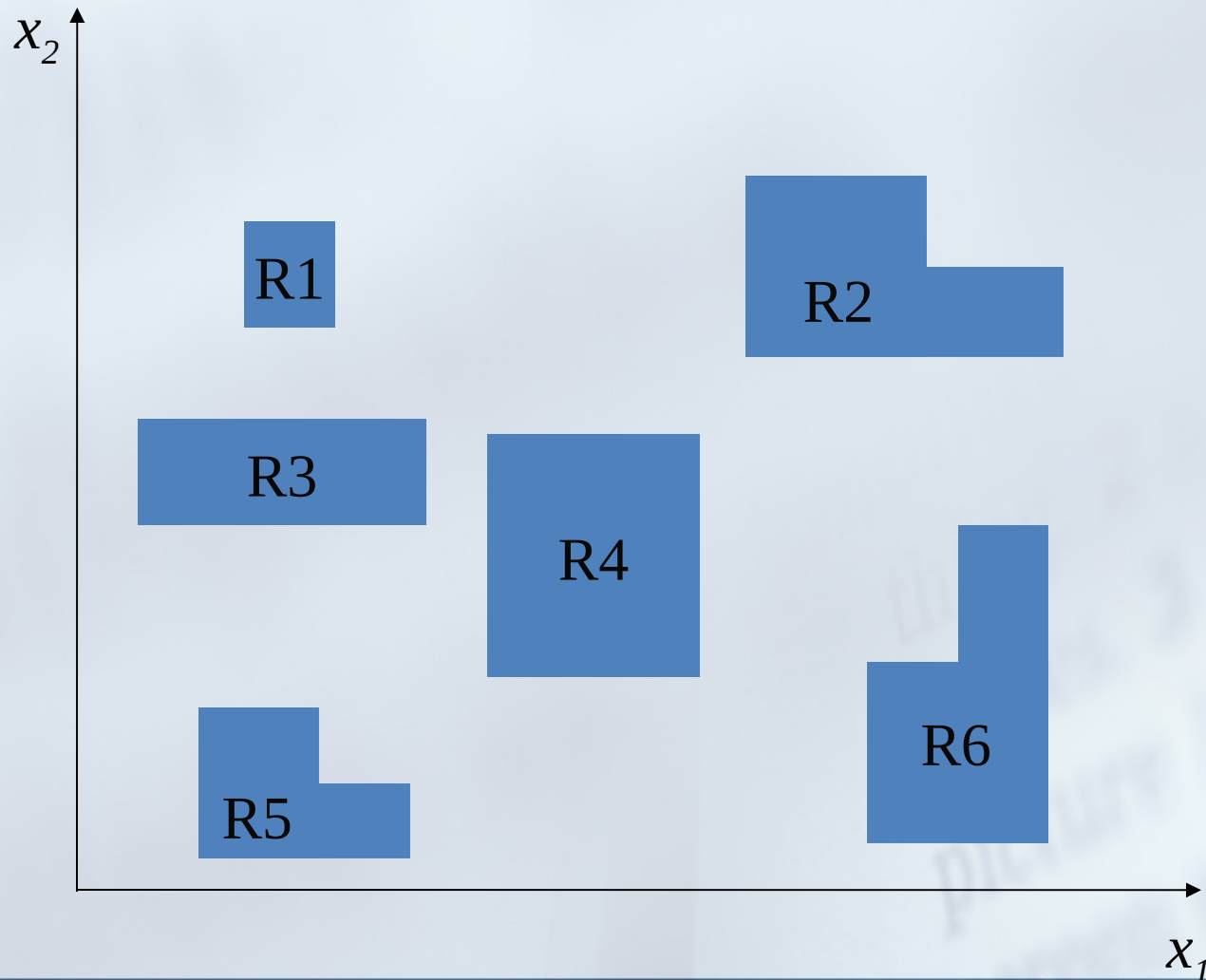
# Momentos

## **Momentos de HU:**

- Característica Global e Invariante
- Medidas puramente estatísticas da distribuição dos pontos.

Considere a imagem binária de um objeto  $M \times N$  onde  $I(x,y)$  representa o estado do pixel  $(x,y)$  – preto ou branco.

# Um Exemplo – Momentos de HU



# Um Exemplo – Momentos de HU

Momento	R1	R2	R3	R4	R5	R6
1	1.67E-01	1.94E-01	2.08E-01	1.67E-01	1.94E-01	1.94E-01
2	0.00E+00	6.53E-03	1.56E-02	0.00E+00	6.53E-03	6.53E-03
3	0.00E+00	1.02E-03	0.00E+00	0.00E+00	1.02E-03	1.02E-03
4	0.00E+00	4.56E+05	0.00E+00	0.00E+00	4.56E+05	4.56E+05
5	0.00E+00	4.25E-09	0.00E+00	0.00E+00	4.25E-09	4.25E-09
6	0.00E+00	1.70E+06	0.00E+00	0.00E+00	1.70E+06	1.70E+06
7	0.00E+00	-8.85E+09	0.00E+00	0.00E+00	-8.85E+09	-8.85E+09

R1 e R4, R2 e R5 são diferentes escalas do mesmo objeto

R6 é a versão rotacionada de R2 e R5

# Um Exemplo – Momentos de HU

Analizando os resultados:

Podemos verificar que os momentos são invariantes a rotação, translação e escala.

Note que R3 é o único objeto diferente, e portanto produz diferentes valores.

# Características Estruturais

Extraem informações da estrutura do padrão.

- Concavidades
- Contornos
- Esqueleto
- Perfil
- Área, Distribuição

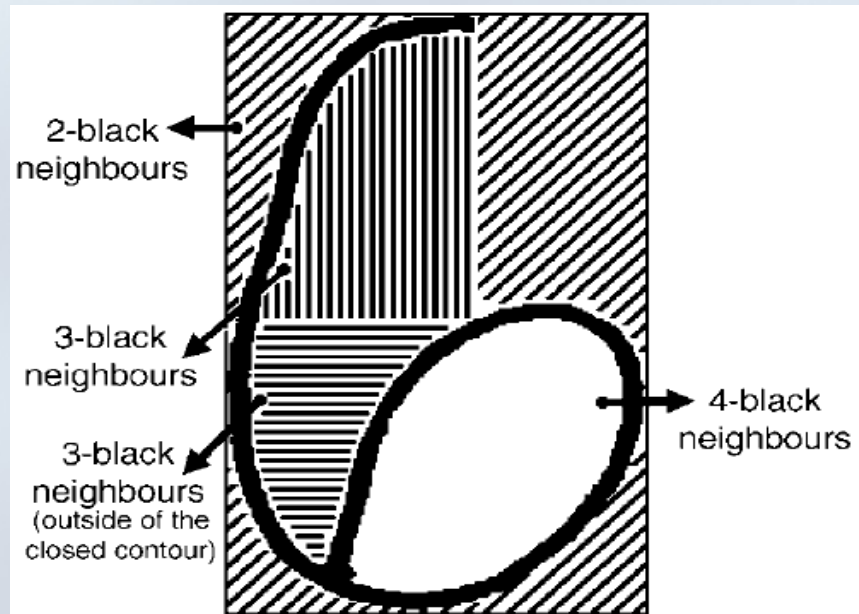
Muitas vezes informações estatísticas são computadas a partir das informações estruturais.



# Concavidades

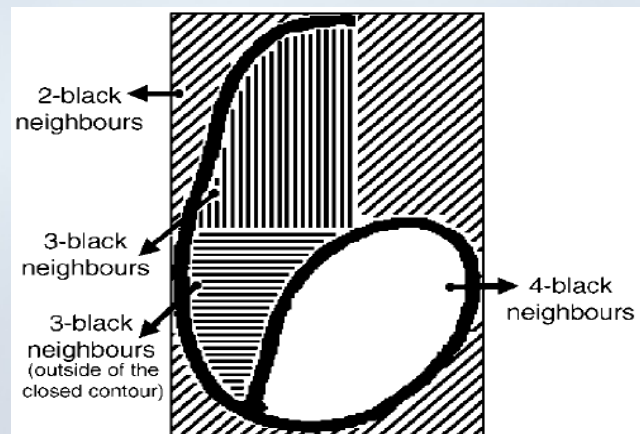
Nesse caso podemos identificar 4 tipos de concavidades

Baseia-se na quantidade de vizinhos pretos

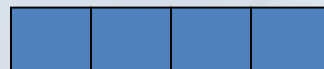


# Concavidades

Como armazenar as informações?



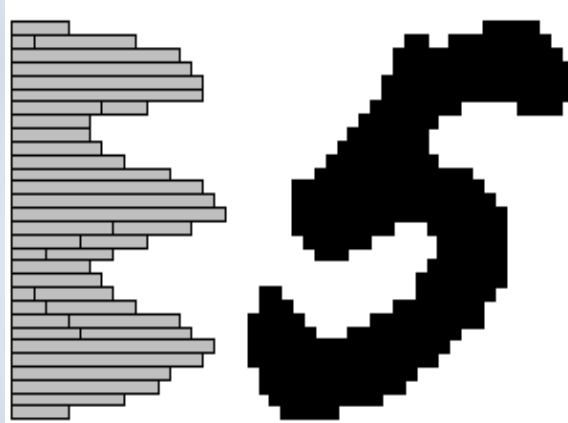
Vetor de características:



Cada posição do vetor corresponde a uma possível configuração. Nesse caso, teríamos um vetor de quatro posições.

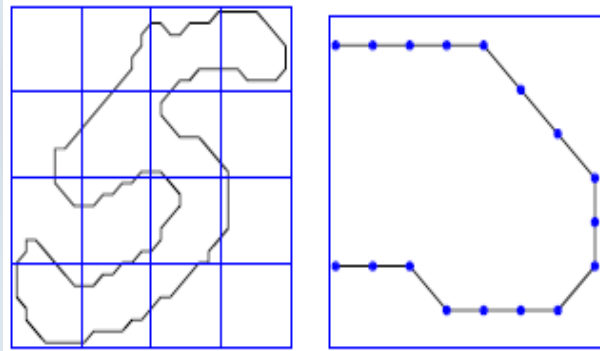
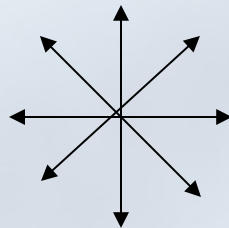
# Distribuição de Pixels

Nesse caso podemos usar um histograma para representar a distribuição dos pixels da imagem.



# Contorno

Para cada pixel do contorno, contabiliza-se a direção do próximo pixel.



Vetor de características teria 8 posições onde cada posição teria a soma das direções.

# Zoneamento

Zoneamento é uma estratégia bastante usada para enfatizar determinadas regiões de um padrão.

Características locais

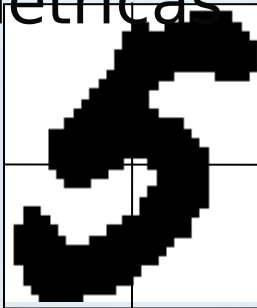
Zonas Simétricas e Assimétricas

Depende do problema que está sendo abordado.

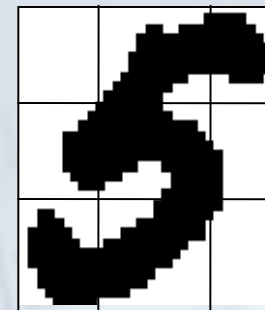
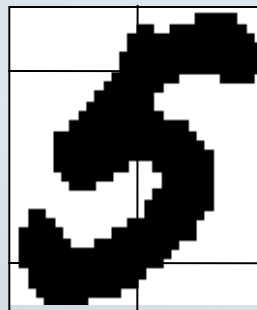
Normalização implícita.

# Zoneamento

4 zonas simétricas



Com base na informação das duas zonas inferiores somente, temos informações similares ao dígito 3



Qual seria a melhor estratégia de zoneamento?



# Mapas de Pixels

Também conhecidos como Edge Maps

Se o objeto puder ser reduzido a um conjunto de linhas horizontais, verticais e diagonais, esses mapas podem fornecer características discriminantes.

Inicialmente a imagem deve ser esqueletizada.

Utiliza simples detectores de linhas

-1	-1	-1
2	2	2
-1	-1	-1

(a)

horizontal

-1	2	-1
-1	2	-1
-1	2	-1

(b)

vertical

-1	-1	2
-1	2	-1
2	-1	-1

(c)

Diagonal 45°

2	-1	-1
-1	2	-1
-1	-1	2

(d)

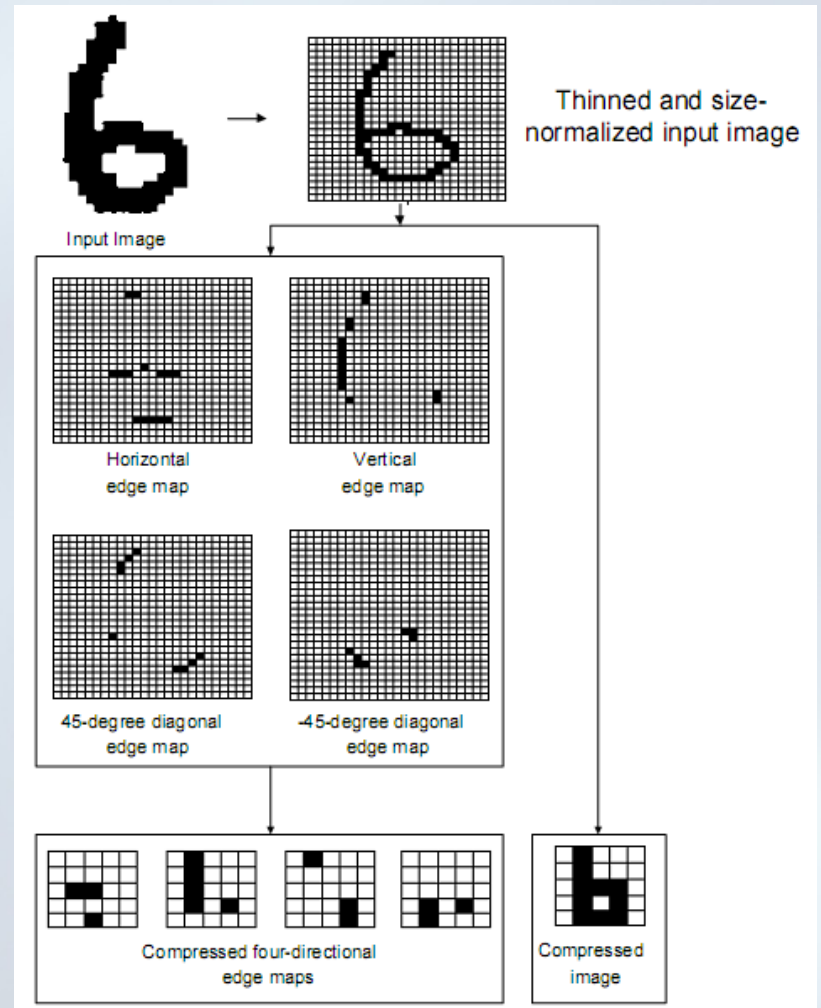
Diagonal -45°

# Mapas de Pixels

Após a detecção das linhas, as mesmas são compactadas em mapas menores

Diminuir custo computacional

Retêm informações mais importantes



# Distâncias

Outra característica com um bom poder de discriminação é a DDD (*Directional Distance Distribution*)

Calcula a distância de cada pixel branco (preto) para seu mais próximo vizinho preto (branco).

Utiliza 8 direções

# Perguntas?

**Dúvidas, Críticas, Sugestões,  
Reclamações?**

[diegobertolini@utfpr.edu.br](mailto:diegobertolini@utfpr.edu.br)