



# Projecte de Final de Carrera

## 2445: Anàlisi i Resolució de Puzzles

Xavier Sureda Rodrigo  
13 de Setembre de 2010

# Índex

1. Introducció
2. Adquisició, Detecció i Segmentació
3. Extracció del Contorn  
i de les Característiques
4. Resolució del puzzle
5. Conclusions

# 1. Introducció

1.1. Objectiu

1.2. Puzzles acceptats

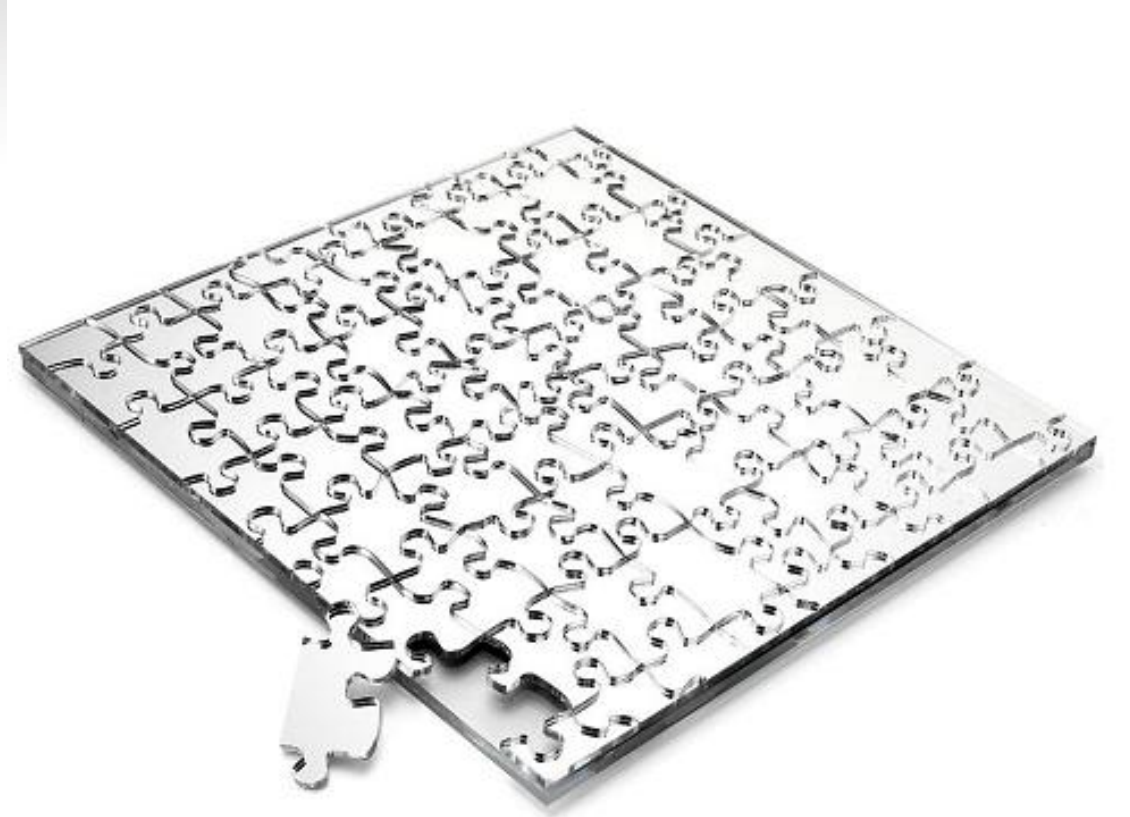
1.3. Puzzles no acceptats

1.4. Sistema de resolució

1. Objectiu | 2. Puzzles Acceptats | 3. Puzzles no Acceptats | 4. Sistema de Resolució

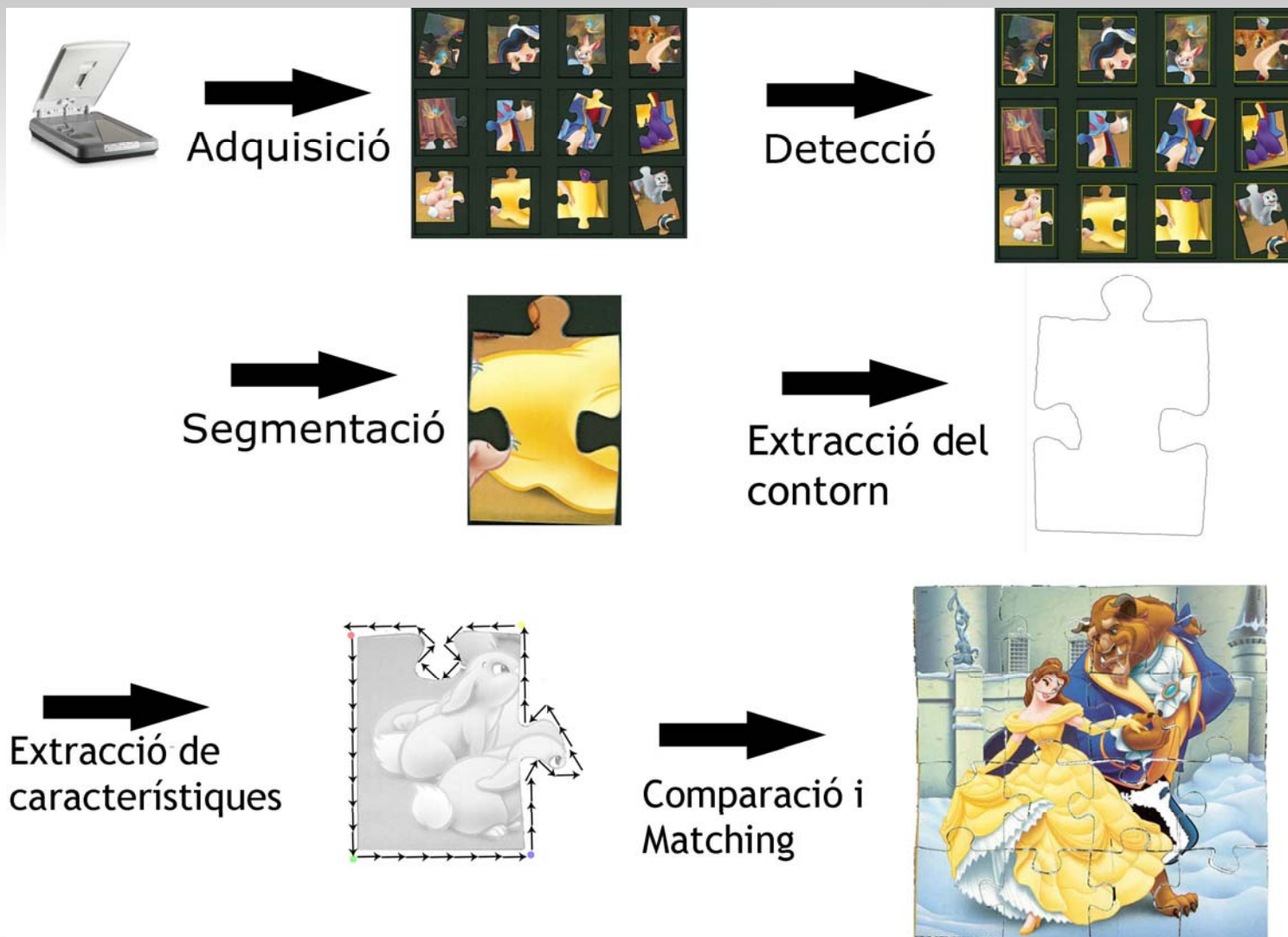
Resoldre un puzzle de manera automàtica, utilitzant eines i algorismes de Visió per Computador







1. Objectiu | 2. Puzzles Acceptats | 3. Puzzles no Acceptats | **4. Sistema de Resolució**





# 2. Adquisició, Detecció i Segmentació

2.1. Mètodes d'adquisició

2.2. Problemes d'adquisició

2.3. Adquisició

2.4. Detecció

2.5. Segmentació



## 1. Mètodes d'Adquisició | 2. Problemes d'adquisició | 3. Adquisició | 4. Detecció | 5. Segmentació

### Càmera fotogràfica:

#### Avantatges:

- Rapidesa

#### Inconvenients:

- Dificultat per crear entorn controlat

### Escàner:

#### Avantatges:

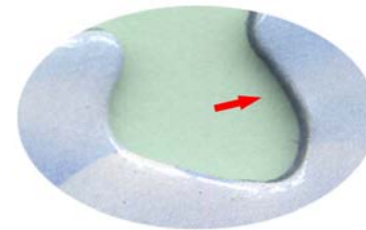
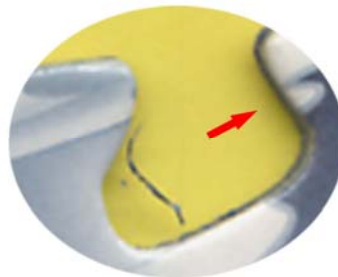
- Luminositat i entorn controlat

#### Inconvenients:

- Soroll i ombres

Opció escollida: **Escàner.**

## El color de fons per minimitzar les ombres



Separació entre peces per a una correcta identificació



1. Mètodes d'Adquisició | 2. Problemes d'adquisició | **3. Adquisició** | 4. Detecció | 5. Segmentació



Per a detectar l'objecte sobre un fons de color homogeni s'ha fet servir el *threshold* **Otsu**





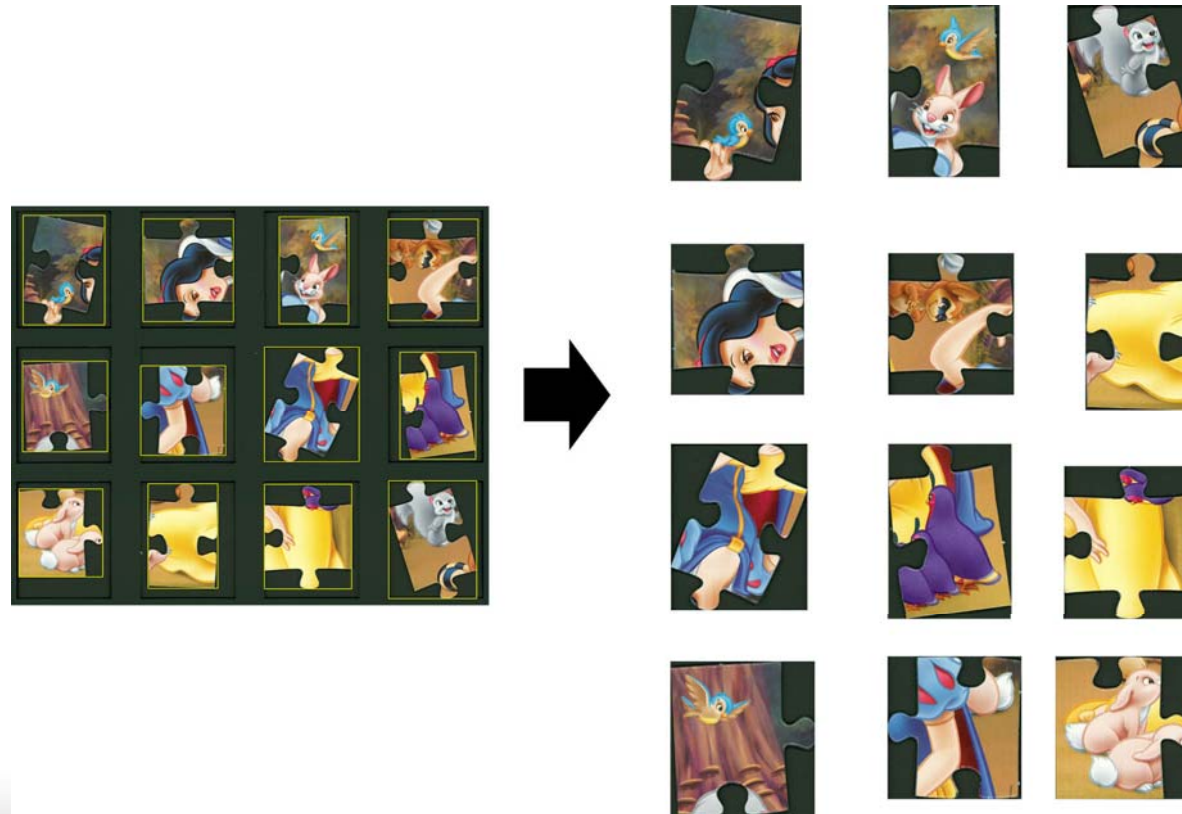
**Bounding Box:** Mínim quadrangle que conté un objecte

**ROI:** Regió d'Interès

Crear una nova imatge a partir de la **ROI** generada sobre el **Bounding Box** de la peça



Passar d'una imatge amb múltiples peces a múltiples imatges amb una sola peça





# 3. Extracció del Contorn i de les Característiques

3.1. Extracció del Contorn

3.2. Tipus d'anàlisi

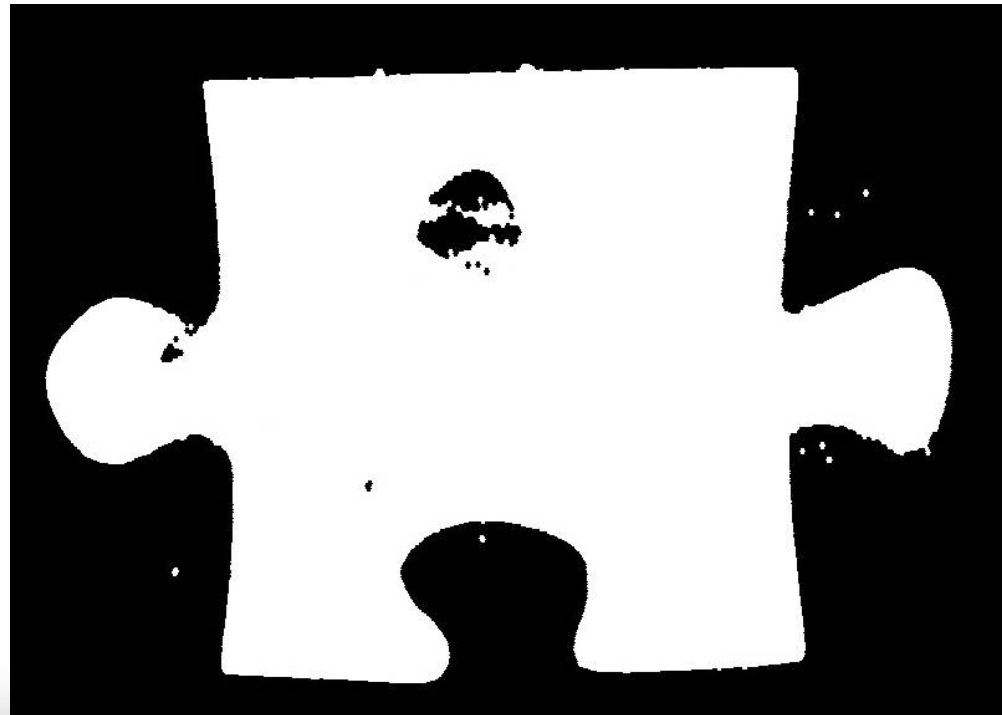
3.3. Obtenció de Costats

3.4. Codificació

3.5. Classificació

**1. Extracció del Contorn** | 2. Tipus d'anàlisi | 3. Obtenció de Costats | 4. Codificació | 5. Classificació

**Binaritzem** la imatge i fem un *Closing* (dilatació + erosió) per eliminar part del soroll extern a l'objecte



## 1. Extracció del Contorn | 2. Tipus d'anàlisi | 3. Obtenció de Costats | 4. Codificació | 5. Classificació

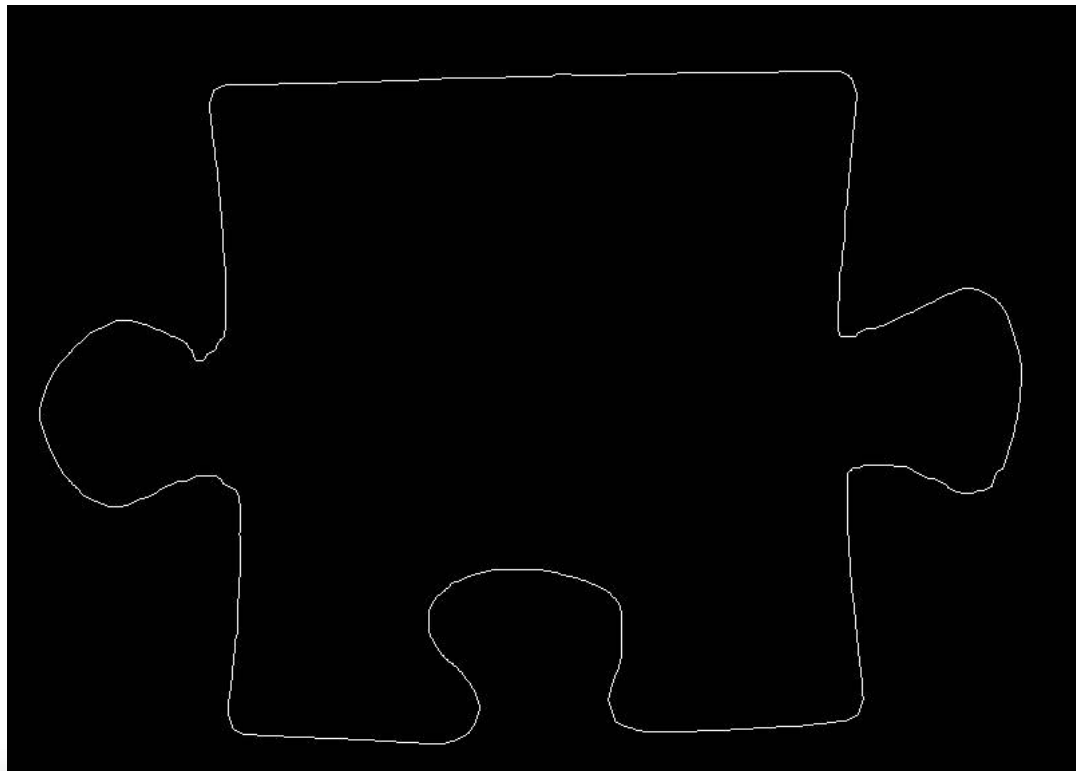
**cvFindContour:** Genera grups de seqüències de punts que ressegueixen contorns.

L'objecte amb la seqüència de punts més gran és la peça.



**1. Extracció del Contorn** | 2. Tipus d'anàlisi | 3. Obtenció de Costats | 4. Codificació | 5. Classificació

**cvMinAreaRect2**: Retorna el centre i l'angle de rotació a aplicar sobre un objecte per tal que, un cop reorientat, el Bounding Box sigui el menor possible



## Per Color:

A favor:

- Avantatges amb imatges molt colorejades

En contra:

- Sovint requereix de pre-anàlisi per forma
- Problemes amb moltes puzzles amb colors molt semblants

## Per Forma:

A favor:

- Avantatges amb imatges amb poca trama o color

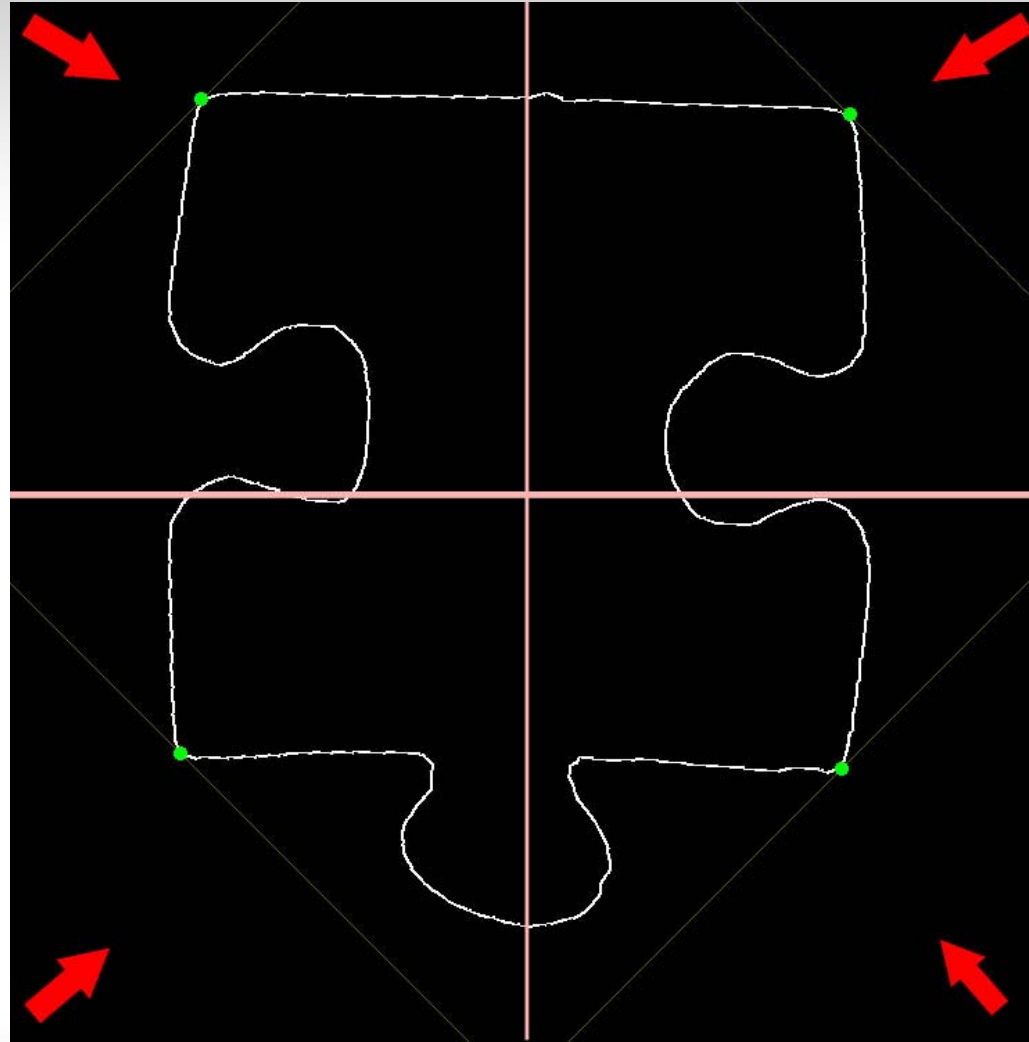
En contra:

- Problemes amb moltes peces amb forma molt semblant.

Opció escollida: **Anàlisi per Forma**

Partint la imatge correctament orientada per la mitat tant horitzontalment com vertical, tenim una cantonada a cada sub-imatge

Fem un escombrat amb una recta diagonal, comencant de l'extrem perpendicular a la recta, fins a trobar el primer punt de l'objecte







Identificació de costats segons la forma:

- *Indent*
- *Outdent*
- *Recta*

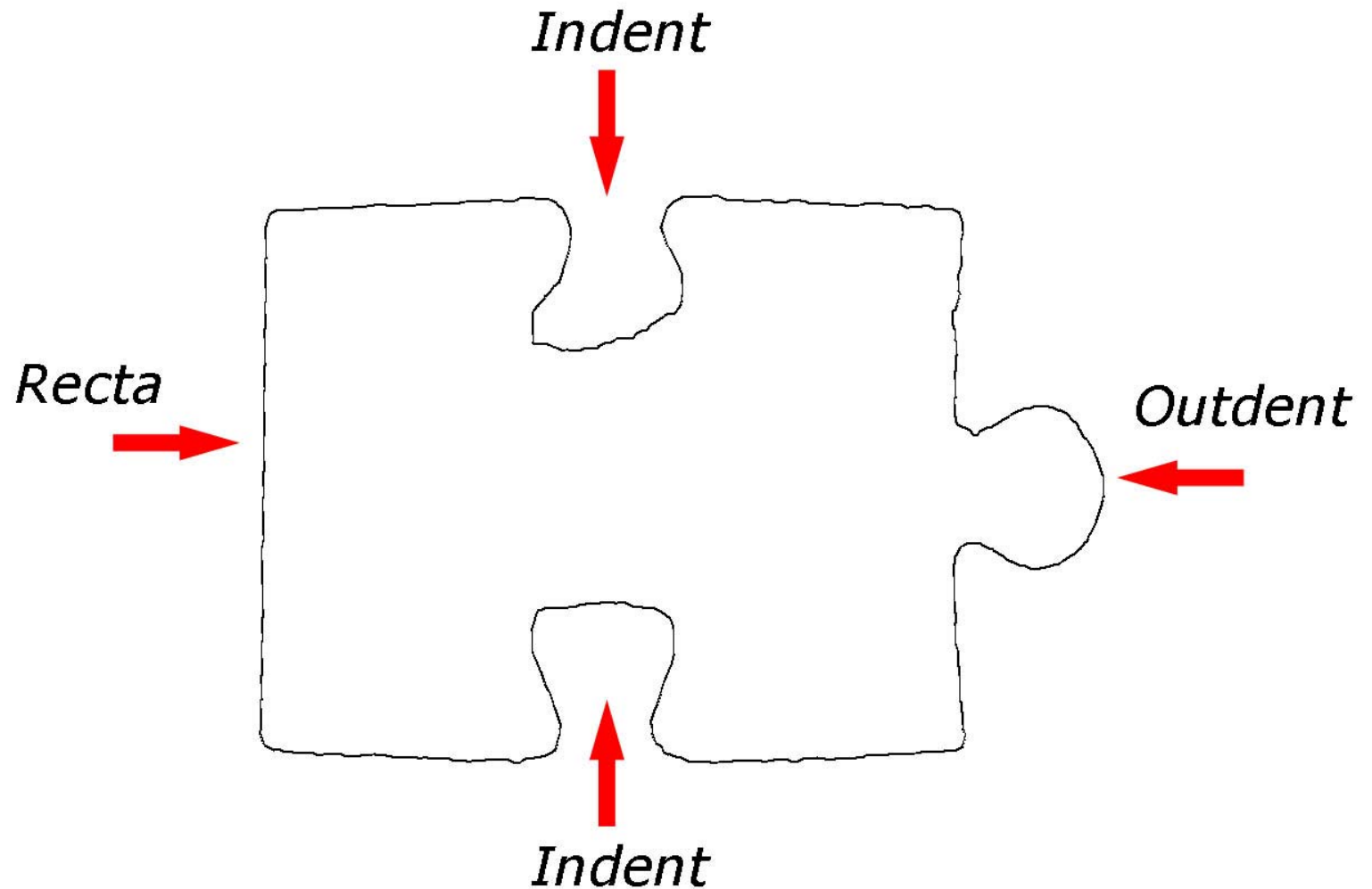
Identificació de la peça segons els costats:

- *Cantonada*
- *Contorn*
- *Centre*

**Indent:** Donats dos punts cantonada, els punts del segment que els uneix tendeixen al centre de l'objecte

**Outdent:** Donats dos punts cantonada, els punts del segment que els uneix tendeixen a l'exterior de l'objecte

**Recta:** Donats dos punts cantonada, els punts del segment que els uneix no tendeixen a cap costat



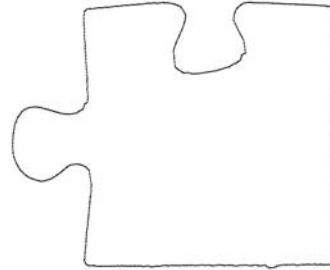
**Cantonada:** Té dos costats tipus *Recta* consecutius i dos costats de qualsevol dels altres tipus

**Contorn:** Té un costat tipus *Recta* i tres costats de qualsevol dels altres tipus

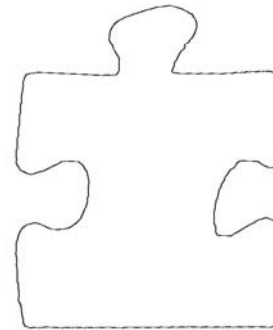
**Centre:** No té cap costat tipus *Recta*

1. Extracció del Contorn | 2. Tipus d'anàlisi | 3. Obtenció de Costats | 4. Codificació | **5. Classificació**

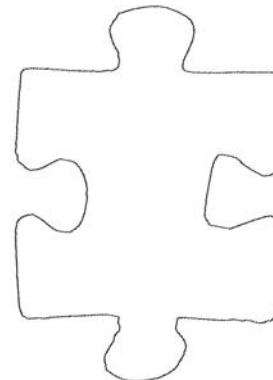
Cantonada:  
1 Outdent  
1 Indent  
2 Rectes



Contorn:  
1 Outdent  
2 Indent  
1 Rectes



Centre:  
2 Outdent  
2 Indent  
0 Rectes



# 4. Resolució del Puzzle

4.1. Procediment

4.2. Consideracions

4.3. Estructura de Dades

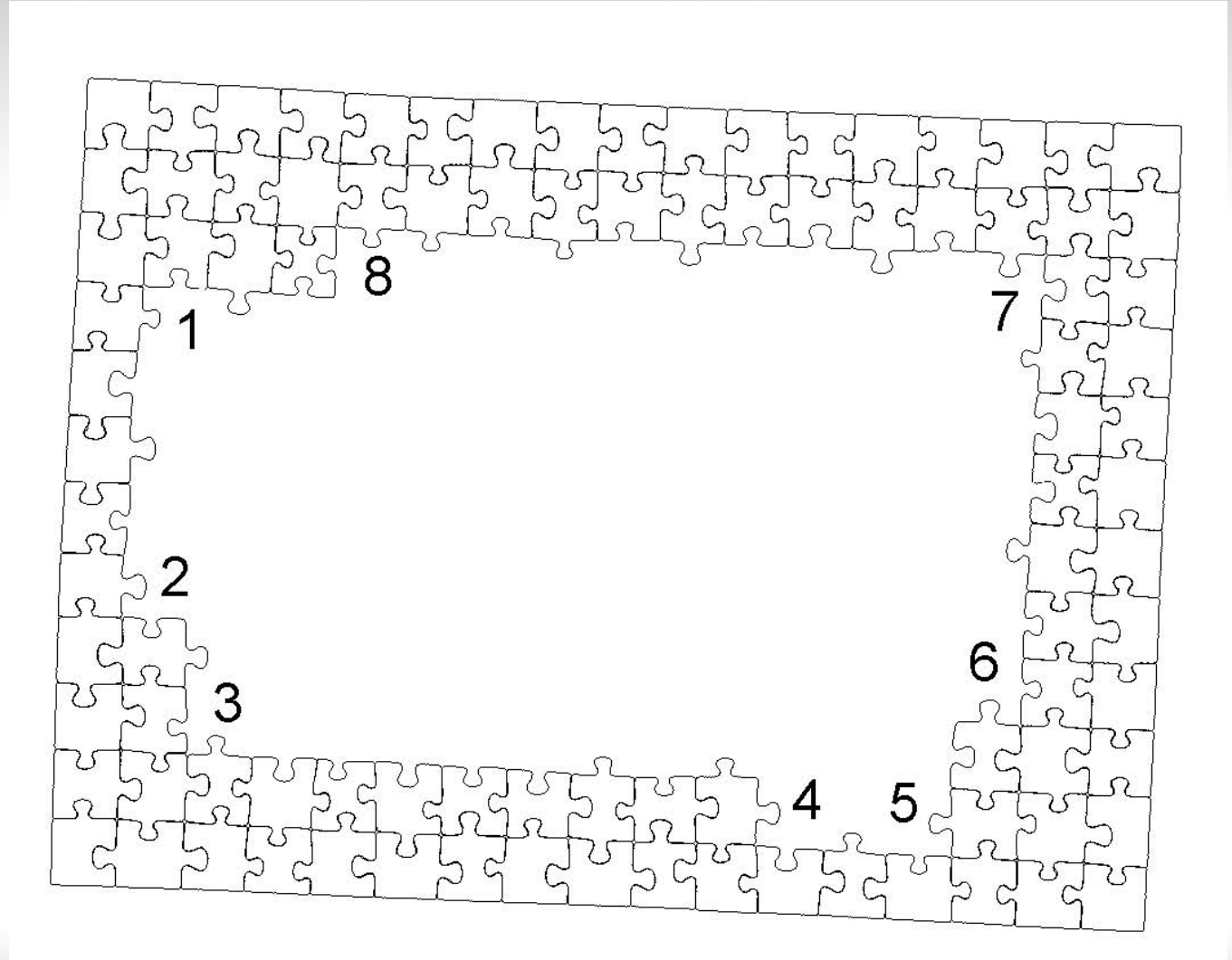
4.4. Comparació

4.5. Procés de Matching

4.6. Resultats



1. Començar per una peça *Cantonada*
2. Comparar peça actual amb les possibles *Contorn i Cantonada*
3. Repetir pas 2 fins aconseguir tot el marc exterior
4. Comparar les peces *Centre* en els extrems on hi ha dos o més costats que han d'estar en correspondència
5. Repetir pas 4 fins ubicar totes les peces



- Comparar costats de tipus contraris (*Indents - Outdents*)
- Comparar costats contigus al costat *Recte* de *Contorn* només amb peces *Contorn* i *Cantonada*
- Els costats *Rectes* de *Contorn* han d'estar alineats
- Dues peces *Cantonada* no poden estar juntes si hi ha peces *Centre*
- Les peces *Centre* s'han de comparar almenys en dos costats

## Peça:

- Ruta del fitxer **imatge** de la peça
- **Angle** de rotació inicial
- **Tipus** de peça
- **Vector** de quatre elements, un per cada costat
  - **Chain Code** del costat
  - **Tipus** del costat

## Mapa de Peces:

- Matriu  $M * N$  amb un **identificador** de cada peça i la orientació.

Algorisme de la **Distància de Levenshtein**: Donades dues cadenes, es dóna el cost de convertir una cadena en l'altra, en funció de la quantitat d'operacions a realitzar. Les operacions són, **insertar** caràcter, **eliminar** caràcter, **modificar** caràcter.

		f	a	s	t
	0	1 1	2 2	3 3	4 4
c	1 1	1 2 2 1	2 3 2 2	3 4 3 3	4 5 4 4
a	2 2	2 2 3 2	1 3 3 1	3 4 2 2	4 5 3 3
t	3 3	3 3 4 3	3 2 4 2	2 3 3 2	2 4 3 2
s	4 4	4 4 5 4	4 3 5 3	2 3 4 2	3 3 3 3

**Variació** de l'algorisme de **Levenshtein**: Una assignació de costos diferent per a cada operació i depenent dels valors de la cadena.

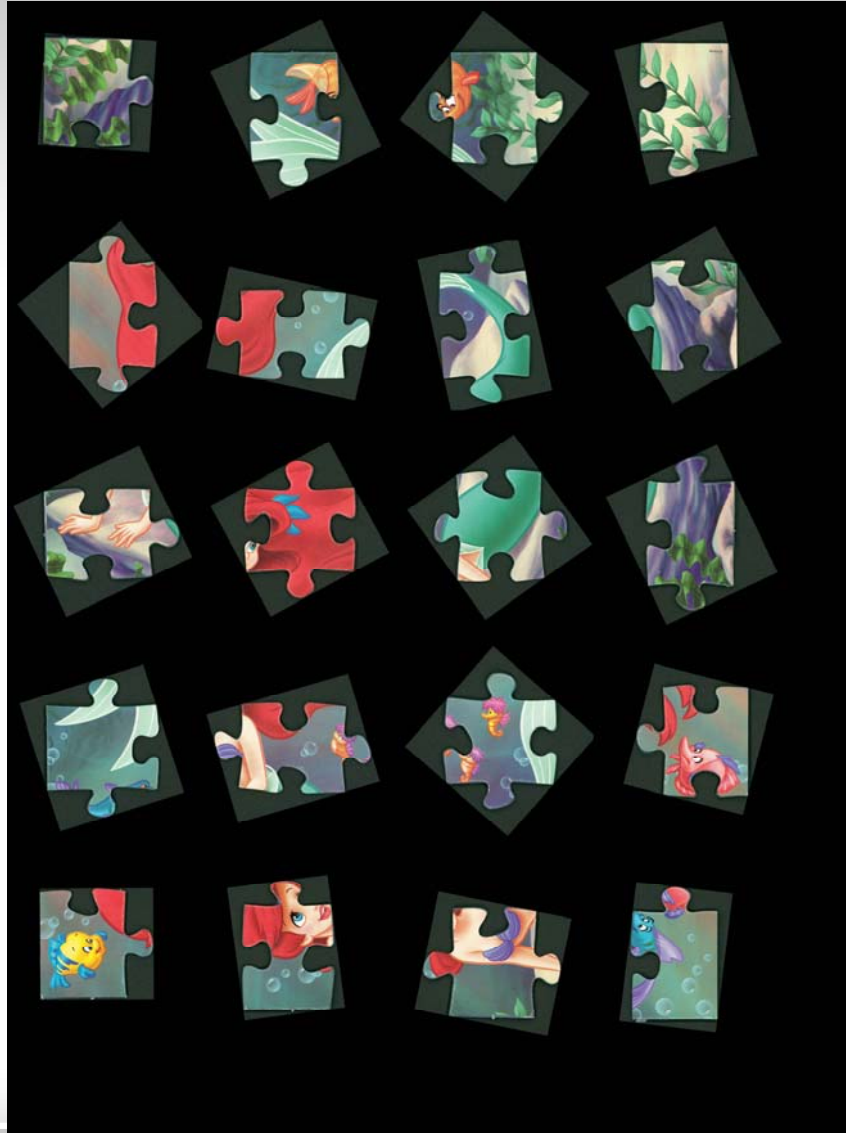
Acció		Cost
Insertar		0.5
Eliminar		0.5
Modificar	$\pm 1$	0.5
	$\pm 2$	1
	$\pm 3$	1
	$\pm 4$	1

- Posar en correspondència aquelles peces que minimitzin el cost
- Acceptar aquell muntatge que minimitza el cost de manera global
- Omplir el Mapa de Peces

Títol	Peces	Temps	Error	Correcte
Blancaneus	12	17.171s	5.50%	Sí
Bella i Bèstia	16	79.687s	4.725%	Sí
Sireneta	20	2258.89s	11.38%	No
Ventafocs	25	--	--	No







# 6. Conclusió

6.1. Conclusió

6.2. Treball futur

## 1. Conclusió | 2. Treball futur

- Fent ús de tècniques de Tractament d'Imatge s'han detectat peces de puzzle, retallat i posat en correspondència
- Aquestes peces responen a una forma concreta i predeterminada
- El color de fons en l'adquisició pot dificultar la detecció
- La falta de característiques dificulta el *Matching*

## 1. Conclusió | **2. Treball futur**

- Afegir anàlisi per color
- Menys restriccions de forma
- Matching a qualsevol punt del contorn de l'objecte
- Reconeixement d'absència d'objectes







**Moltes**

**Gràcies**