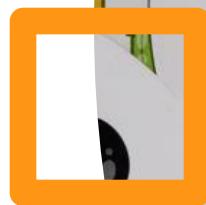


Feature extraction and classification project

Mirabelle ABDULMASSIH,
Nathan CABASSO,
Jiayi HAO,
Ferdinand MOM

IMAGE 2023



Problem Summary

Monitor Dobble Games



Dobble Game



The objective of Dobble is to win points by finding out the unique symbol shared by two cards.


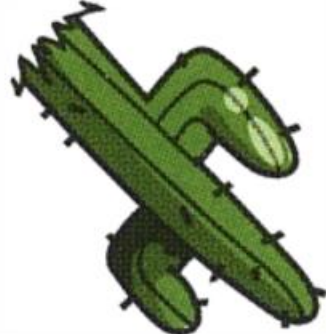
All players play at the same time. A player must always be the fastest to locate the identical symbol between 2 maps, name it out loud, then take the card, put it down or discard it.

Monitor Dobble Games

The specific goal of this project is to design a symbol classifier for Dobble symbols, which must:

- accept an image of a symbol crop as input,
- and return the class of the symbol as output.

This classifier will be used as part of an automatic game monitor to tell when players are cheating.

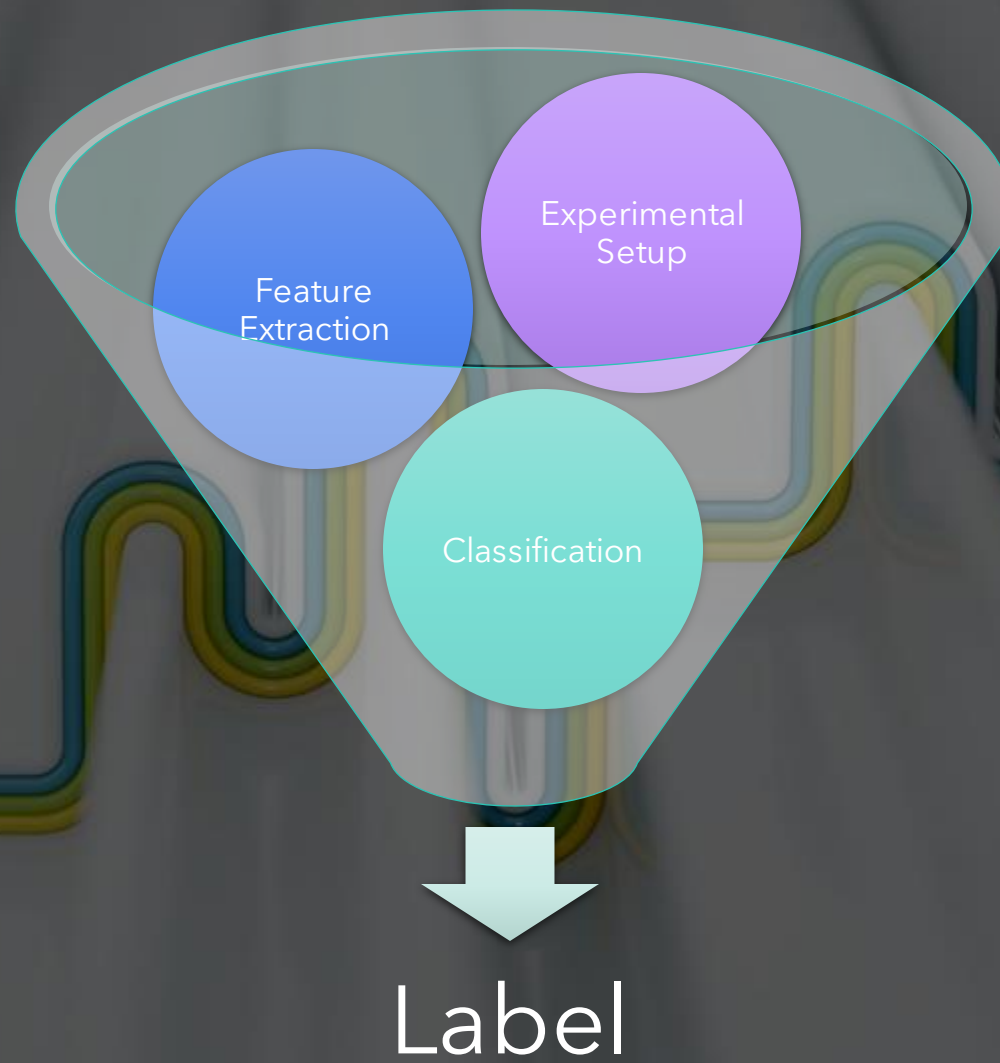
Sample input	Expected output
	15 (Symbol code for “Dragon” class)
	27 (Symbol code for “Cactus” class)



Pipeline Overview

Three Main Stages

Pipeline Overview





Feature
Extraction

ental
o

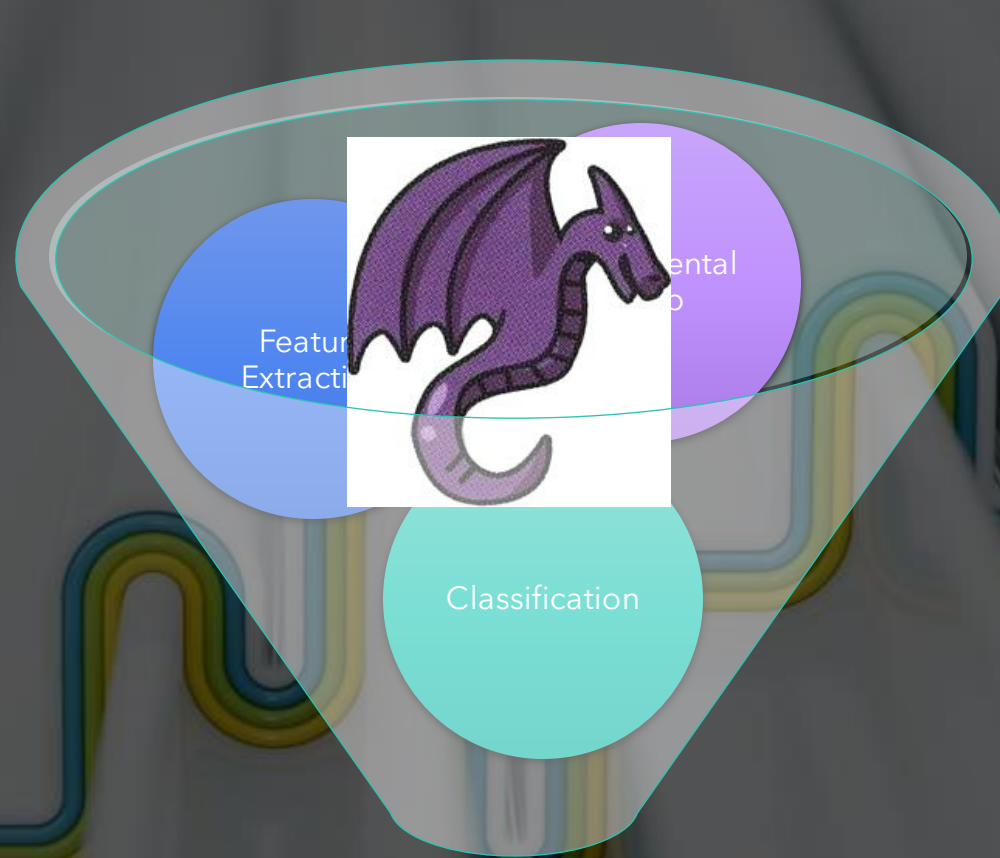
Classification



Label

Pipeline Overview

Pipeline Overview



Label

15: 'dragon'



Feature Extraction

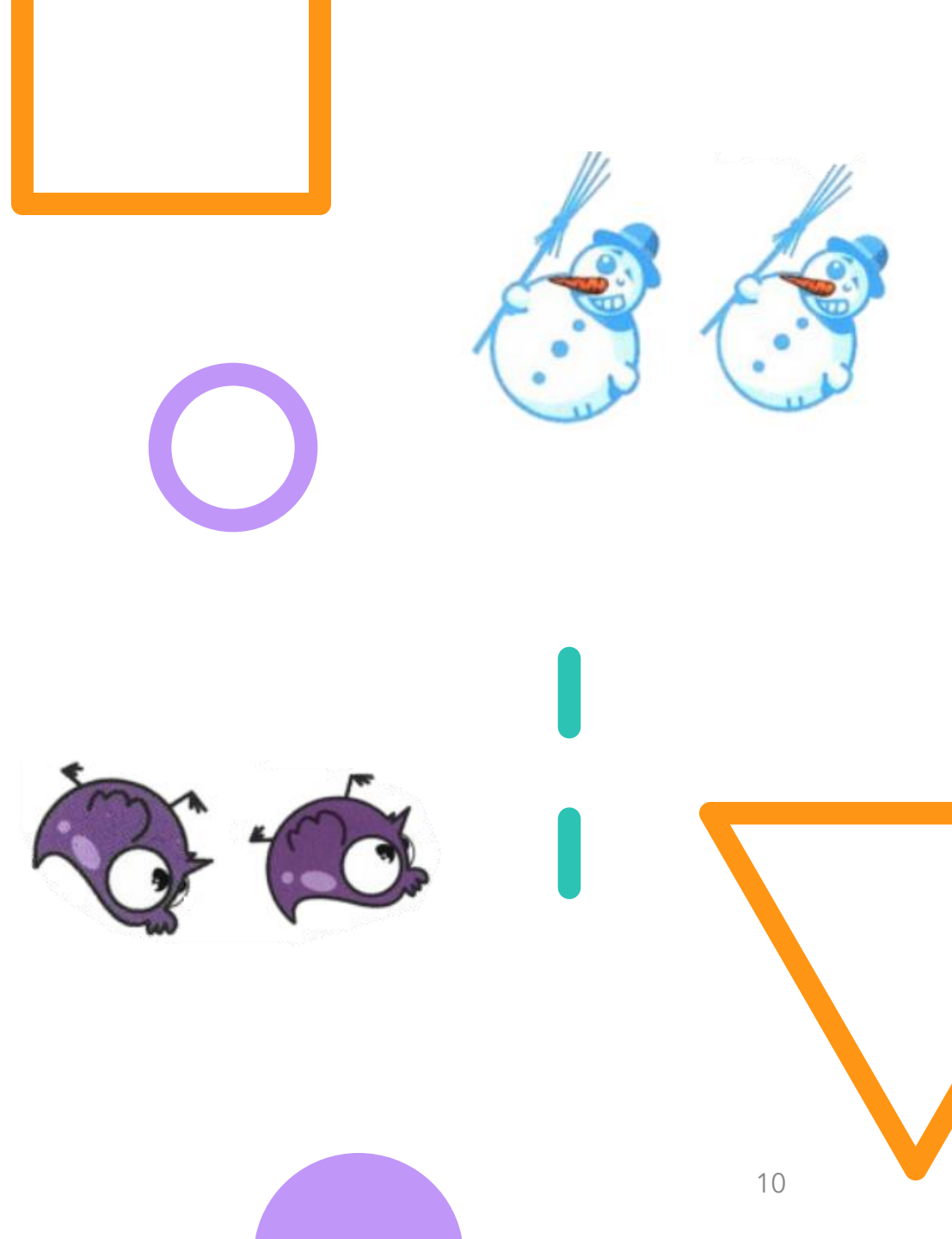
Descriptors



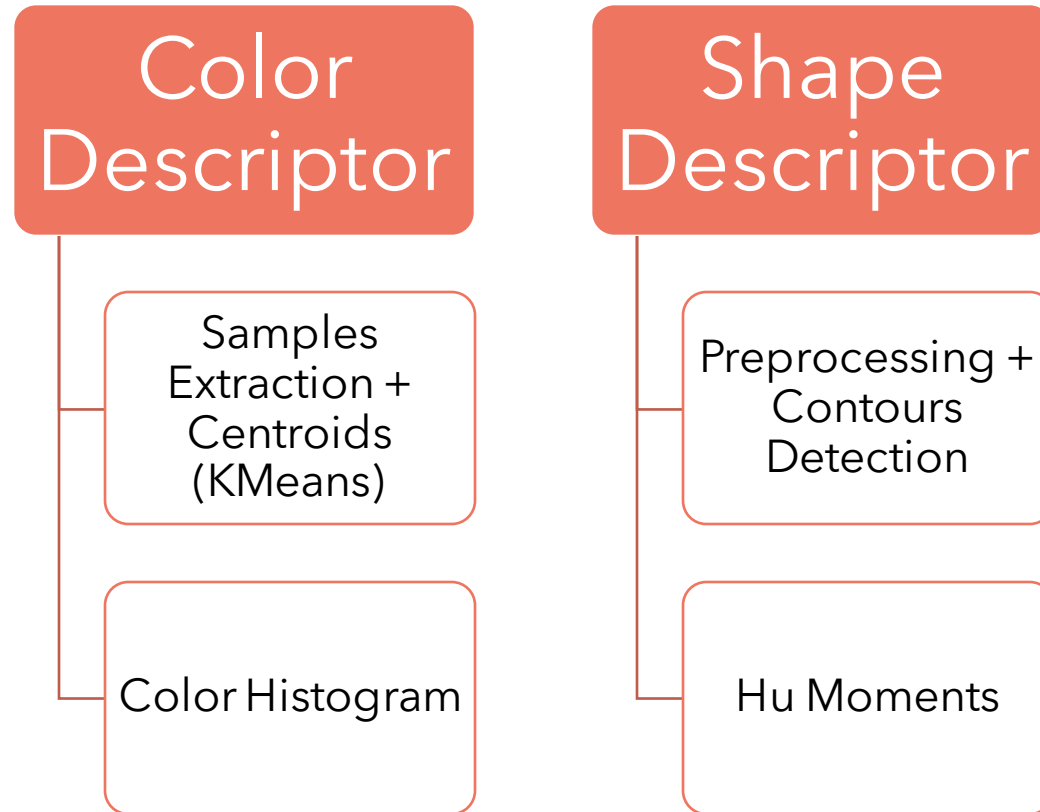
Feature Extraction: Requirements

The features must be

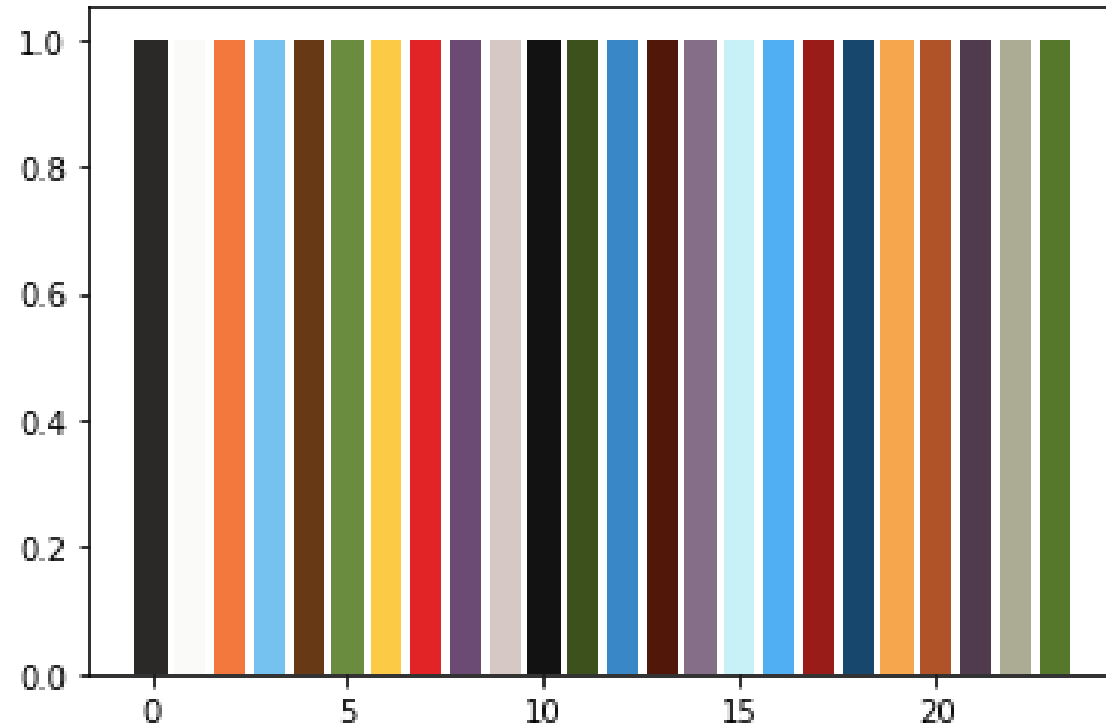
- scale-invariant
- rotation-invariant
- noise-invariant



Feature Requirements: **Descriptors**



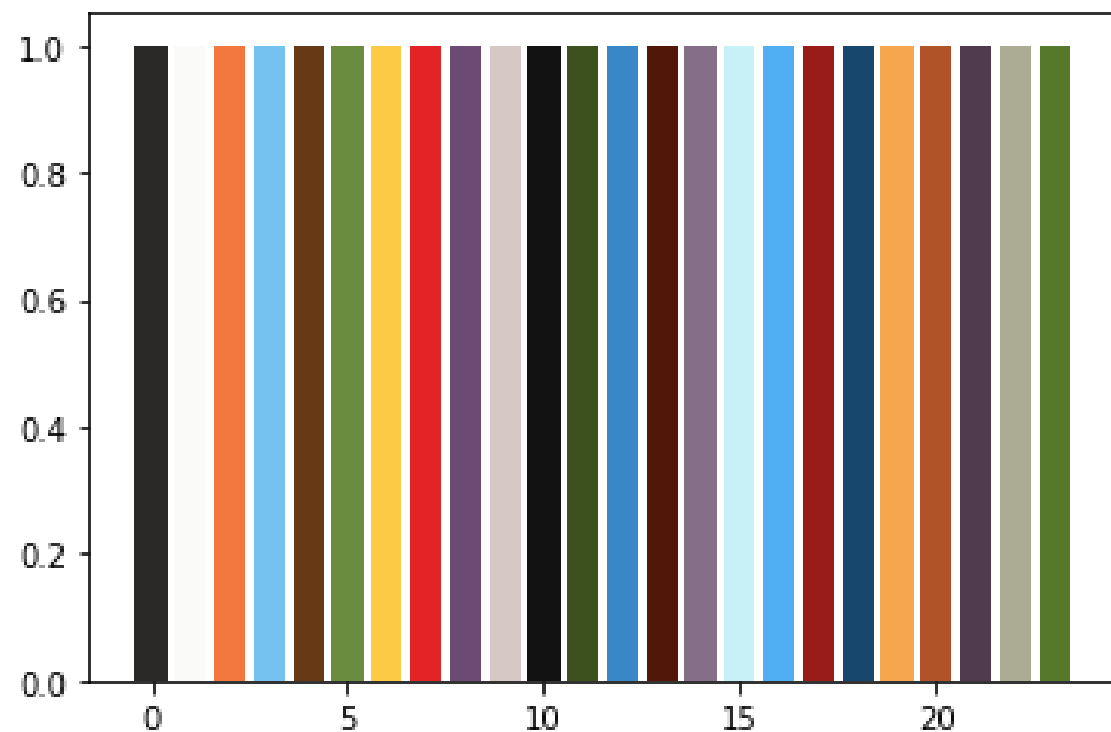
Feature Requirements: Descriptors



Color
Descriptor

Samples
Extraction +
Centroids
(KMeans)

Color
Histogram



Feature Requirements: Descriptors

Color
Descriptor

Samples
Extraction +
Centroids
(KMeans)

Color
Histogram

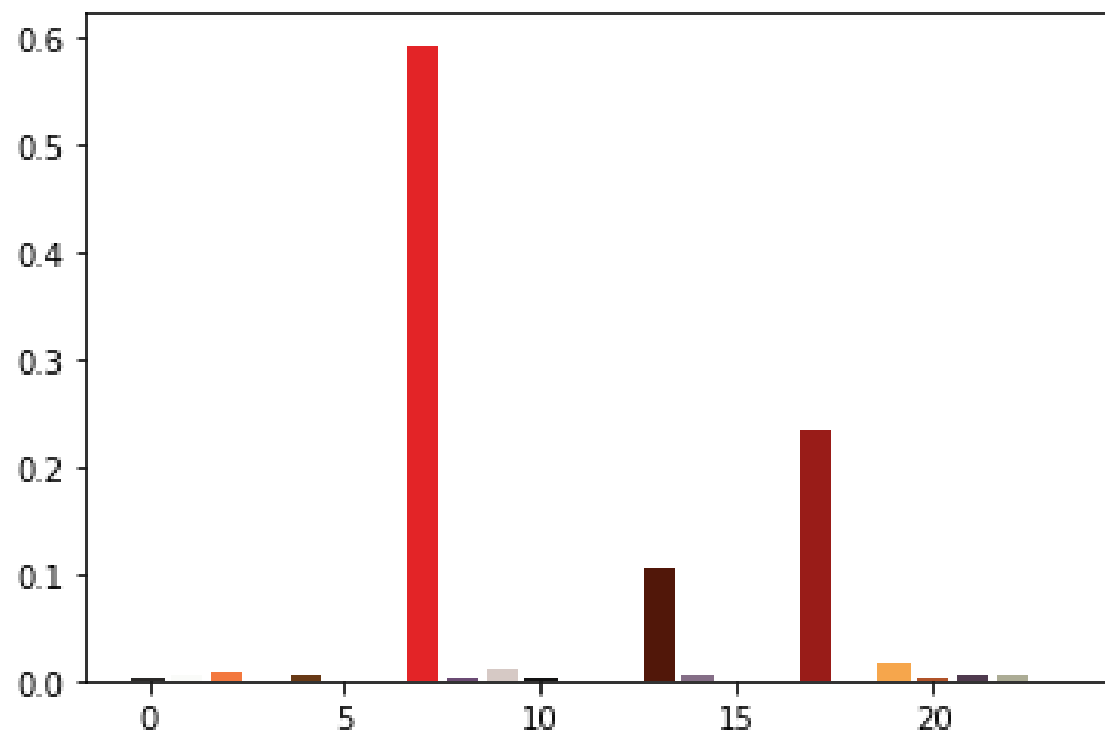


Feature Requirements: Descriptors

Color
Descriptor

Samples
Extraction +
Centroids
(KMeans)

Color
Histogram



Feature Requirements: Descriptors



Color
Descriptor

Samples
Extraction +
Centroids
(KMeans)

Color
Histogram

Feature Requirements: **Descriptors**



Shape
Descriptor

Preprocessing
+ Contours
Detection

Hu Moments

Feature Requirements: **Descriptors**



Shape
Descriptor

Preprocessing
+ Contours
Detection

Hu Moments

Feature Requirements: **Descriptors**

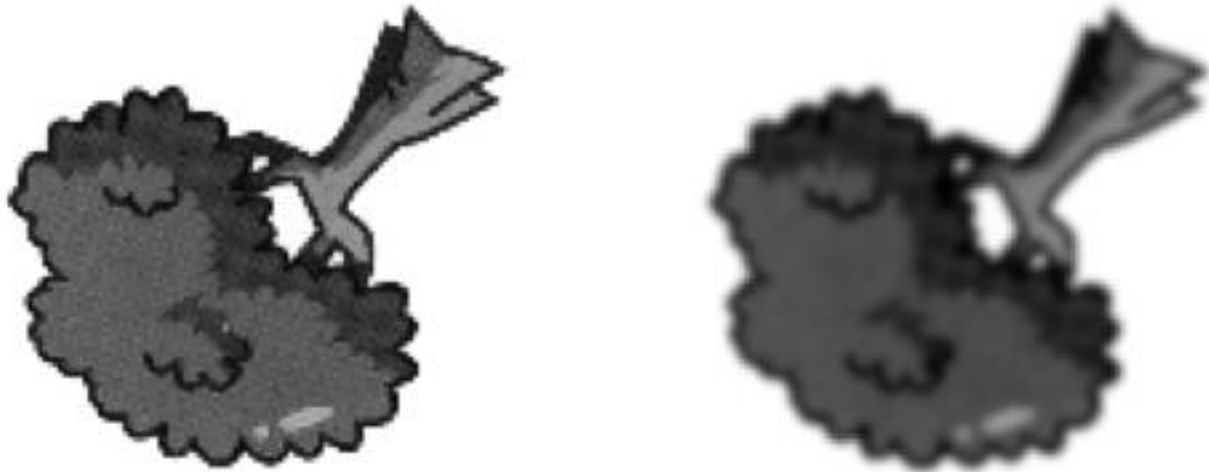


Shape
Descriptor

Preprocessing
+ Contours
Detection

Hu Moments

Feature Requirements: **Descriptors**



Shape
Descriptor

Preprocessing
+ Contours
Detection

Hu Moments

Feature Requirements: **Descriptors**



Shape
Descriptor

Preprocessing
+ Contours
Detection

Hu Moments

Feature Requirements: **Descriptors**

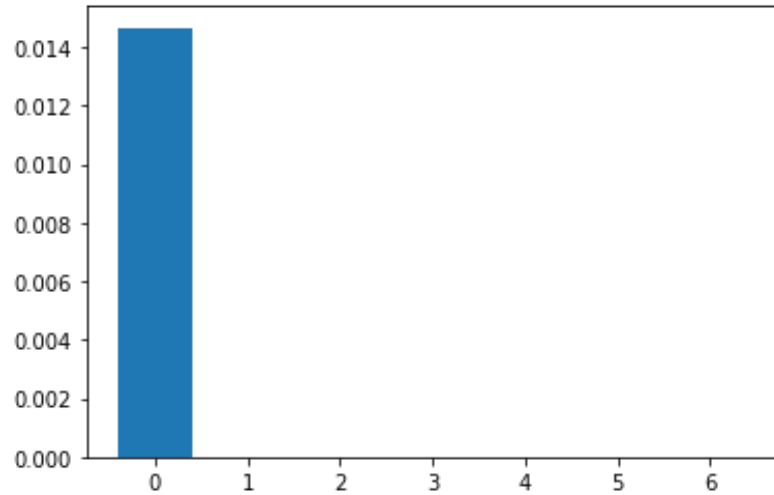


Shape Descriptor

Preprocessing
+ Contours
Detection

Hu Moments

Feature Requirements: **Descriptors**

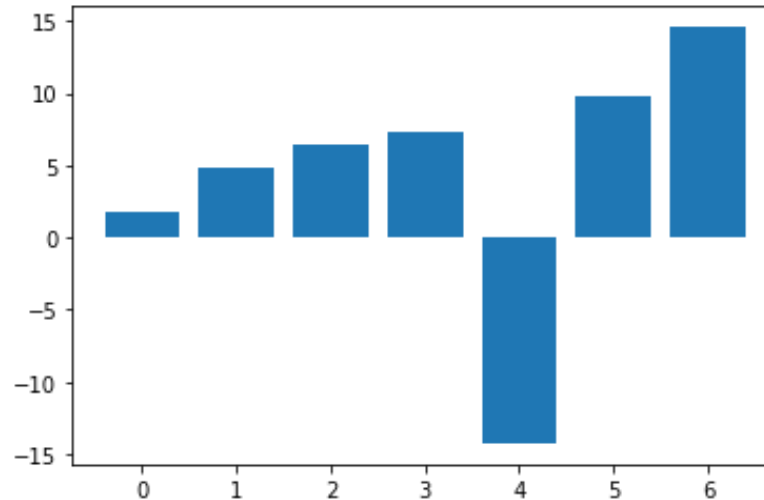
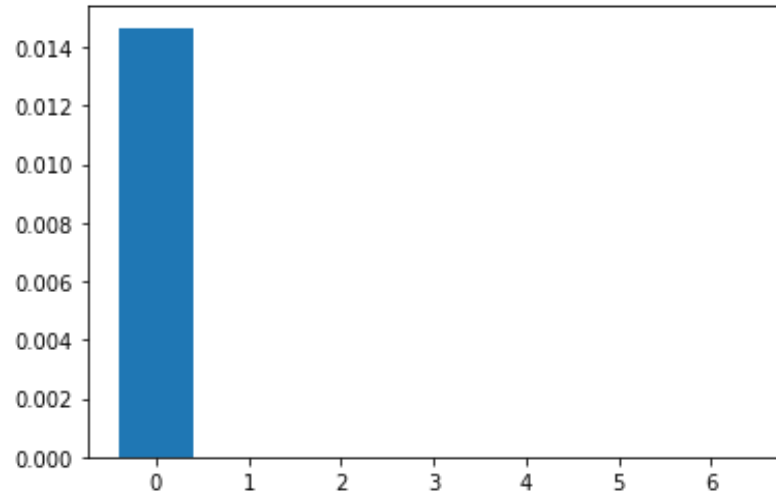


Shape
Descriptor

Preprocessing
+ Contours
Detection

Hu Moments

Feature Requirements: Descriptors



Shape
Descriptor

Preprocessing
+ Contours
Detection

Hu Moments

A large teal circle is centered on the slide. To its left, there are several short, teal, dashed lines of varying lengths, arranged in a curved pattern. At the bottom right of the teal circle, there is a small, solid purple circle.

Classification

Using scikit-learn



Classifiers Used

Dummy

Random class

1/57 accuracy

Linear

Linear **SVM**

Non-
Linear

SVM: polynomial, RBF

KNearest-neighbor: *Euclidian* and *cosine* distance

Decision Tree

Random Forest

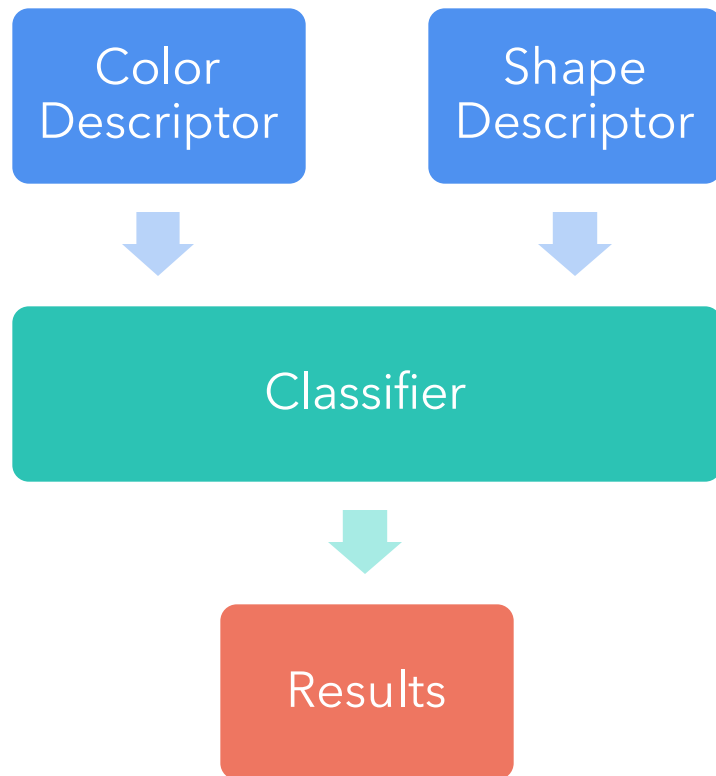


Fusion

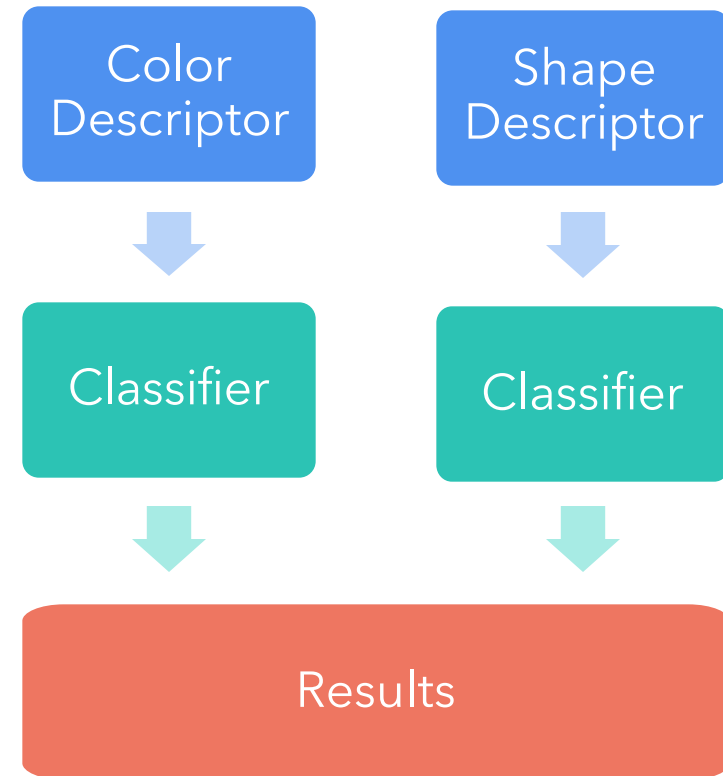
Explicit only

Fusion

Early Fusion



Late Fusion

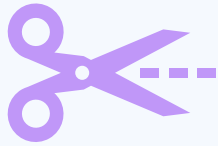


The background of the slide features a close-up, artistic photograph of various pieces of laboratory glassware, including Erlenmeyer flasks, test tubes, and a graduated cylinder, some containing liquids. The lighting is dramatic, with strong highlights and shadows. Overlaid on the left side of the image is a large, semi-transparent white circle. To the right of this circle is a smaller, solid blue circle. Several short, teal-colored dashed lines are scattered in the upper left quadrant of the slide.

Experimental Setup

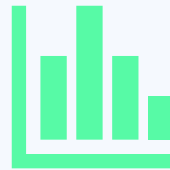
Creating an objective environment
and fair metrics to identify
the best model

Setup



Train Test Split: Validation Set

Randomized
Stratified



Data Augmentation

Added random **rotations**
Added random **scale** changes



Metrics

Accuracy: number of correct prediction /
total test

Confusion Matrix

Execution time

Setup



Metrics

Accuracy: number of correct prediction / total test

Confusion Matrix

Execution time



Results

Summary and Benchmark

Benchmark and Summary

	Type de fusion	Classifiers utilisés	Temps d'execution (s)	Meilleur accuracy (%)	Autres accuracy (%)
Data aug = 10 CV = 40	Early	Knn	26,403	100	95,41
		SVM linear			98,83
		SVM poly			31,50
		SVM rbf			12,30
		Random forest			100
		Decision tree			99,75
	Late	Knn - Knn	68,033	100	100
		Knn - Random forest			100
		Knn - Decision tree			99,92
		Random forest - Random forest			100
		Random forest - Knn			96,41
		Random forest - Decision tree			99,16
		Decision tree - Decision tree			99,67
		Decision tree - Knn			99,67
		Decision tree - Random forest			99,33
	Stacking early	Decision tree + Random forest	91,98	100	99,67
		Decision tree + pipeline			100
	Stacking late	Decision tree + Random forest	53,77	100	100

Benchmark and Summary

	Type de fusion	Classifiers utilisés	Temps d'execution (s)	Meilleur accuracy (%)	Autres accuracy (%)
Data aug = 5 CV = 40	Early	Knn	12,88	100	92,5
		SVM linear			98,41
		SVM poly			23,12
		SVM rbf			15,16
		Random forest			100
		Decision tree			99,51
	Late	Knn - Knn	33,977	100	99,84
		Knn - Random forest			100
		Knn - Decision tree			100
		Random forest - Random forest			100
		Random forest - Knn			95,06
		Random forest - Decision tree			96,49
		Decision tree - Decision tree			99,68
		Decision tree - Knn			99,52
		Decision tree - Random forest			99,68
	Stacking early	Decision tree + Random forest	46,344	100	99,36
		Decision tree + pipeline			100
	Stacking late	Decision tree + Random forest	23,348	99,84	99,84

Benchmark and Summary

	Type de fusion	Classifieurs utilisés	Temps d'execution (s)	Meilleur accuracy (%)	Autres accuracy (%)
Data aug = 1 CV = 12	Early	Knn	5,185	100	80,7
		SVM linear			84,21
		SVM poly			12,28
		SVM rbf			16,37
		Random forest			100
		Decision tree			97,08
	Late	Knn - Knn	14,97	100	99,42
		Knn - Random forest			99,42
		Knn - Decision tree			99,42
		Random forest - Random forest			100
		Random forest - Knn			91,23
		Random forest - Decision tree			94,15
		Decision tree - Decision tree			98,25
		Decision tree - Knn			95,32
		Decision tree - Random forest			95,91
	Stacking early	Decision tree + Random forest	10,128	100	98,83
		Decision tree + pipeline			100
	Stacking late	Decision tree + Random forest	8,291	100	100

Benchmark and Summary

	Type de fusion	Classifiers utilisés	Temps d'execution (s)	Meilleur accuracy (%)	Autres accuracy (%)
Data aug = 1 CV = 5	Early	Knn	4,648	100	80,7
		SVM linear			84,21
		SVM poly			12,28
		SVM rbf			16,37
		Random forest			100
		Decision tree			97,08
	Late	Knn - Knn	10,124	100	99,42
		Knn - Random forest			99,42
		Knn - Decision tree			99,42
		Random forest - Random forest			100
		Random forest - Knn			91,23
		Random forest - Decision tree			94,15
		Decision tree - Decision tree			98,25
		Decision tree - Knn			95,32
		Decision tree - Random forest			95,91
	Stacking early	Decision tree + Random forest	6,157	100	98,83
		Decision tree + pipeline			100
	Stacking late	Decision tree + Random forest	4,354	100	100

Benchmark and Summary

	Type de fusion	Classifiers utilisés	Temps d'execution (s)	Meilleur accuracy (%)	Autres accuracy (%)
Data aug = 0 CV = 4	Early	Knn	1,657	100	66,67
		SVM linear			64,91
		SVM poly			42,11
		SVM rbf			42,11
		Random forest			100
		Decision tree			92,47
	Late	Knn - Knn	5,028	100	94,74
		Knn - Random forest			92,98
		Knn - Decision tree			92,98
		Random forest - Random forest			100
		Random forest - Knn			73,68
		Random forest - Decision tree			80,70
		Decision tree - Decision tree			87,72
		Decision tree - Knn			89,47
		Decision tree - Random forest			87,72
	Stacking early	Decision tree + Random forest	2,418	100	87,72
		Decision tree + pipeline			100
	Stacking late	Decision tree + Random forest	2,081	92,98	92,98



Conclusion

Thank you!