

# Image Classification hands-on

Wafia Adouane  
Gothenburg University

LT2318: Artificial Intelligence: Cognitive Systems, HT 2018

November 22, 2018

# Outline

Image classification

How humans classify an image?

Features

Detection of features: why and how?

I- Feature detection using OpenCV

SIFT Features

II- Image classification with deep learning

- pre-trained models
- API (even easier!)

# Image classification

## Image classification

# Image classification

- Finding reliable similarities between images that belong to the same category or represent the same object.

# Image classification

- Finding reliable similarities between images that belong to the same category or represent the same object.
- The basic idea is to consider image classification as grouping images into semantic classes based on some features.

How humans classify an image?



# How humans classify an image?

- Looking for specific patterns or specific features which are unique, which can be easily tracked, which can be easily compared.



# How humans classify an image?

- Looking for specific patterns or specific features which are unique, which can be easily tracked, which can be easily compared.
- Well, it is difficult to say how humans find these features. It is already programmed in our brain?

# How humans classify an image?

- Looking for specific patterns or specific features which are unique, which can be easily tracked, which can be easily compared.
- Well, it is difficult to say how humans find these features. It is already programmed in our brain?
- But if we look deep into some pictures and search for different patterns, we will find something interesting.

# Features

- A set of information which can be anything about an object depending on the task to solve.
- They can be about the color, size, center, average color, orientation, minimum and maximum intensity, etc. of the object.

# Types of features

# 1. Visual features

- Invariant representative features (information) which are able to distinguish an image from other images.
- These features should be invariant to different image transformations such as rotation, illumination, scale, viewpoint, noise,... etc.

## 2. Lexical features/Semantic classes

The label we assign to each image for instance: cats, dogs, etc.

Detection of features: why and how?

- We need to convert these visual features (vectors) into a human-readable format (a.k.a. semantic classes).
- This is done by assigning an appropriate semantic class based on the presence of certain features.

```
[0.04738562091503268, ....0.00016651104886399005,
0.00023498288204170558 ..... 0.01682695300342359,
0.0056691565515094924, ....., 0.0, 0.0, 0.0, 0.0, 0.0]
```

```
(array([[ 91., 27., 2., ..., 1., 0., 1.],
       [ 4., 3., 4., ..., 1., 0., 4.],
       [ 2., 1., 1., ..., 1., 6., 3.], ...,
       [ 0., 0., 0., ..., 0., 0., 0.],
       [ 0., 0., 0., ..., 0., 0., 0.]], dtype=float32),)
```



# I- Detection of visual features

- We use OpenCV and SimpleCV libraries to detect invariant representative features.
- **1. Color information:** Hue and HSV histograms
- **2. Geometry features:** structural Analysis and Shape Descriptors, which are mainly related to Edges, SIFT, SURF, Contours, Skeleton, Canny Edge and Laplacian features, etc.

# SIFT Features

- SIFT : Scale Invariant Image Transform
- Features which are based on extracting invariant feature descriptors.
- They are applied to grayscale images (ignore color).
- They are commonly used in CV.

# I- Feature detection (demo)

(Adapted from Mehdi's OpenCV demo, ESLP course 2015)

# Clustering SIFT features

- SIFT features: a vector of vectors and each image has a different number of descriptors.
- We need to cluster/flatten them to create bag of visual words (to be able to compare them)
- Creation of visual words by clustering descriptors of each image based on some similarity matrix (Bruni et al., 2014).

# Lazy Learning

- Instance based learning or Just-in-time learning (Cybenko, 1996): compares the features of an image to the features of each image in the dataset.
- Use the Fast Library for Approximate Nearest Neighbors (FLANN).

## II- Image classification with deep learning (demo)

# 1- Pre-trained models

Explore existing resources and models  
and adapt them to suit your needs.

## 2. API's call

An even easier way is to use ready models using API's

<https://www.clarifai.com>

<https://imagga.com>



# References I

- Bruni Elia, Khanh Tran Nam and Baroni Marco. 2014. Multimodal Distributional Semantics. In the Journal of Artificial Intelligence Research 49 (2014) 1-47.
- Lowe, D. G. (1999). Object recognition from local scale-invariant features. In Computer vision, 1999. The proceedings of the seventh IEEE international conference on, Volume 2, pp. 1150–1157. IEEE.
- Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. International journal of computer vision 60(2), 91–110.
- Muja, M. and D. G. Lowe (2009). Fast approximate nearest neighbors with automatic algorithm configuration. VISAPP (1) 2(331–340), 2.
- Szummer Martin and Picard Rosalind W., 1998. Indoor and Outdoor Image Classification. *IEEE*, pp. 42-51.

# References II

## **Python tutorials:**

OpenCV:

<https://opencv-python-tutroals.readthedocs.org/en/latest/index.html>

SimpleCV:

<http://simplecv.readthedocs.org/en/1.0/>

## **Image Processing guide:**

<http://homepages.inf.ed.ac.uk/rbf/HIPR2/wksheets.htm>