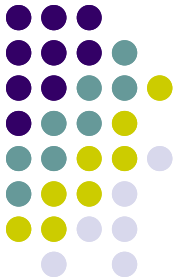
Scalable Image Search with Deep Image Representation



Viet Nguyen

viet.nguyen@tum.de

Friday, 9th February 2018

Introduction

Context-based Image Retrieval system

Database

Challenges

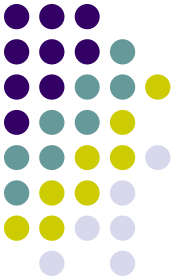
- **Database**: Large dynamic set of unlabeled images
  - How to manage to add a new image to the database?
  - How to build a model from unlabeled data?
- **Response time**:
  - Fast response (*regardless of database's size*)

Dataset

- **Web Gallery of Art: https://www.wga.hu/**
  - Painting
  - Sculpture
  - ceramics
  - graphics
  - …





Model

Pipeline

VGG16

Feature extractions

CAM extractions

Sum pooling

PCA + l_2-norm.

Aggregation

Database

Cosine similarity

Top *N* results

Database

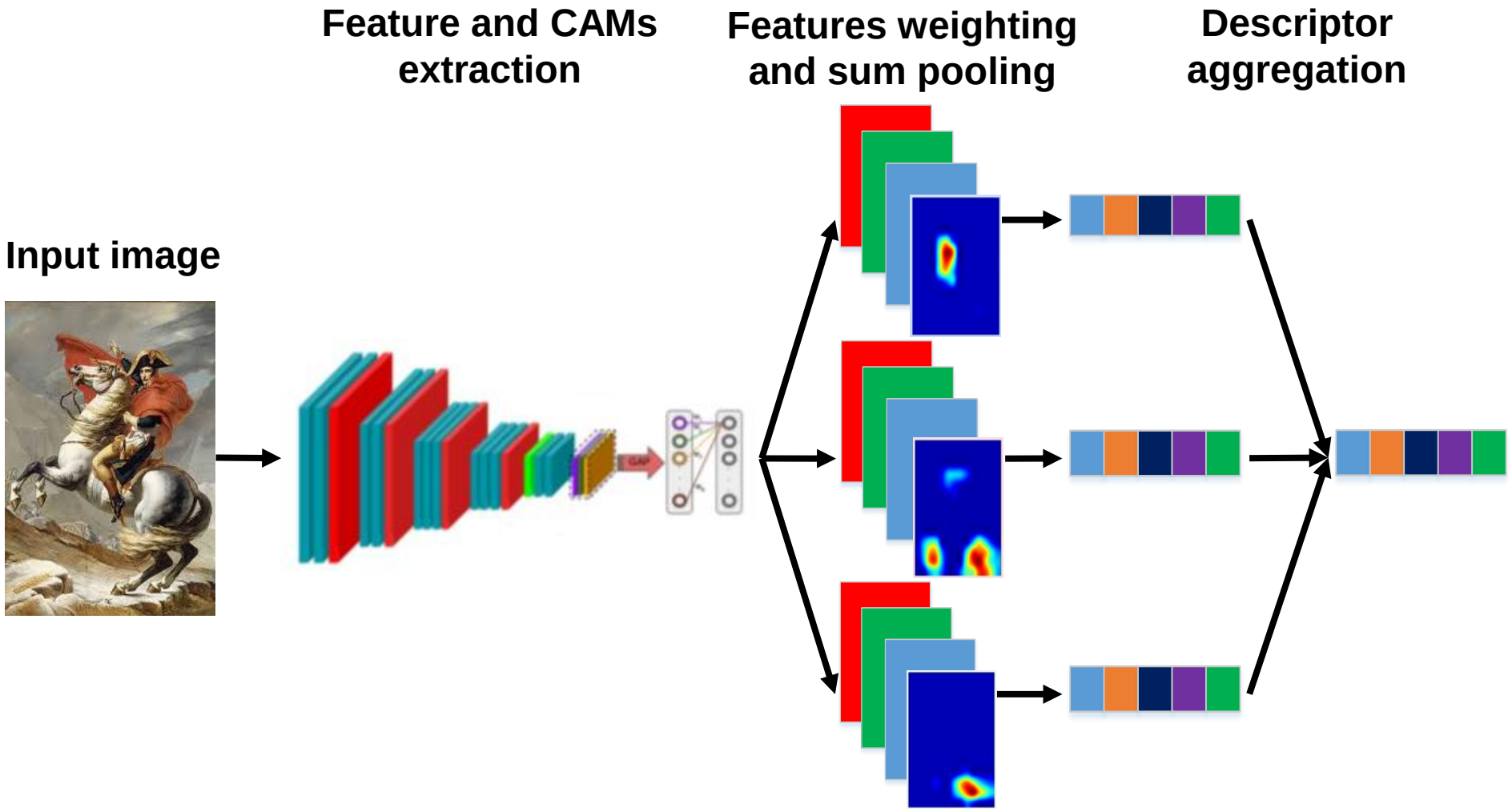
Feature descriptors

Feature matching

Feature descriptor

The most similar image

Architecture



Class Activation Maps

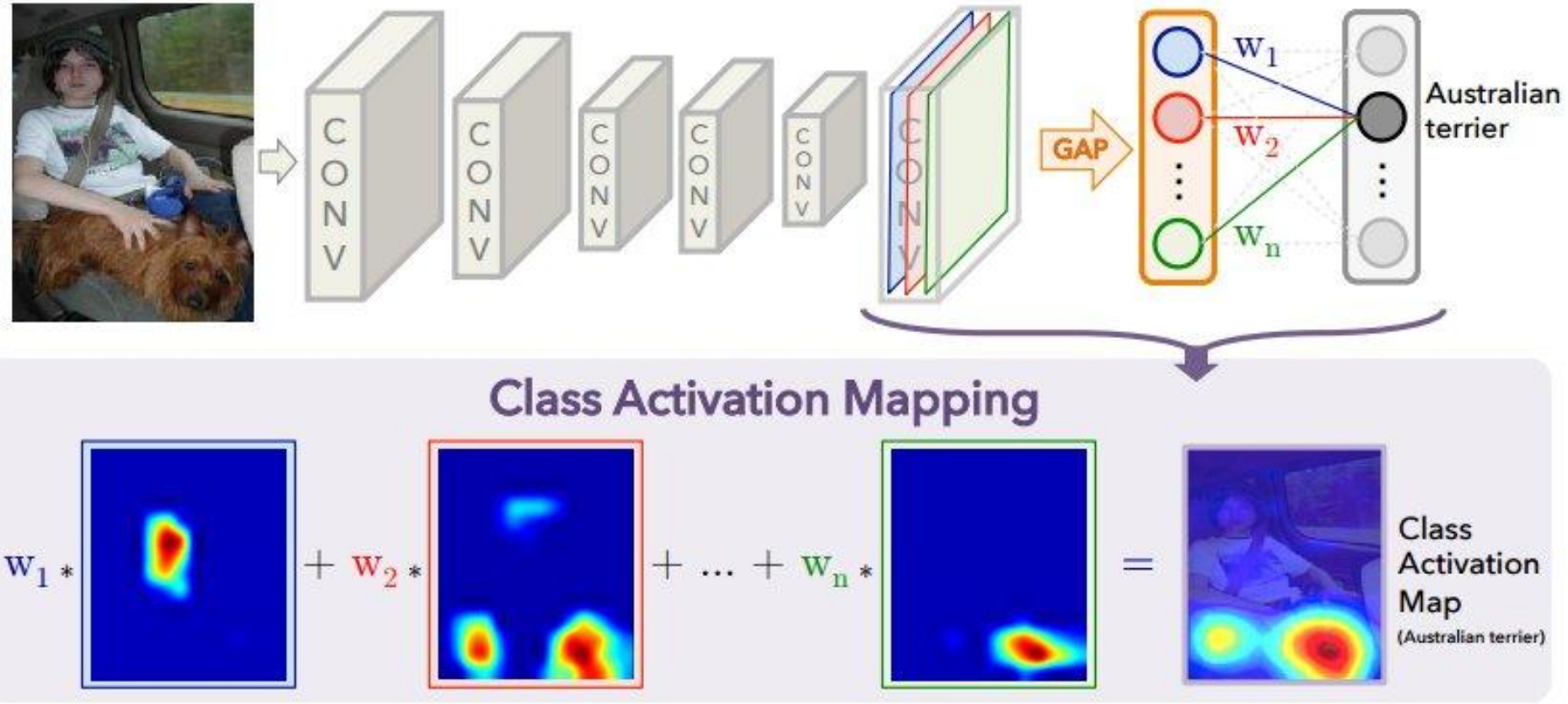


Image is taken from https://github.com/metalbubble/CAM

Class Activation Maps

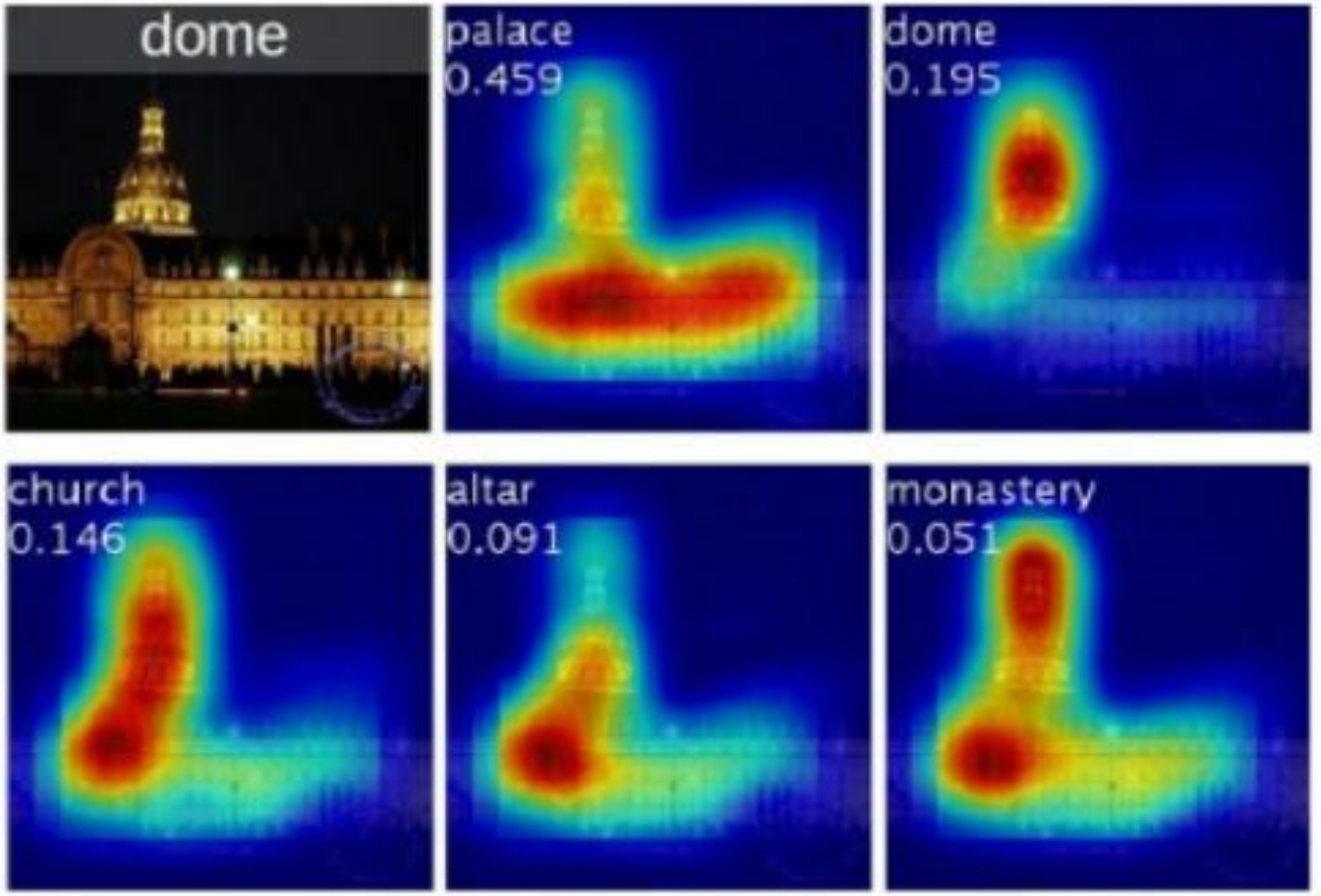


Image is taken from https://github.com/metalbubble/CAM

Cosine similarity

- α is angle between two feature vectors
  X = $(x_1, x_2, \ldots, x_n)$ and Y = $(y_1, y_2, \ldots, y_n)$

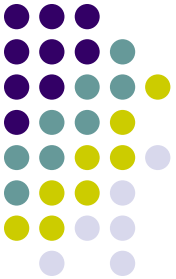
$$CS(X,Y) := \cos(\alpha) = \frac{X * Y}{|X| * |Y|} = \frac{\sum_i x_i y_i}{\sqrt{\sum_i x_i^2}\sqrt{\sum_i y_i^2}}$$

CS = 0.9456

CS = 0.8056

CS = 0.0912

Feature matching



ORB: A fusion of

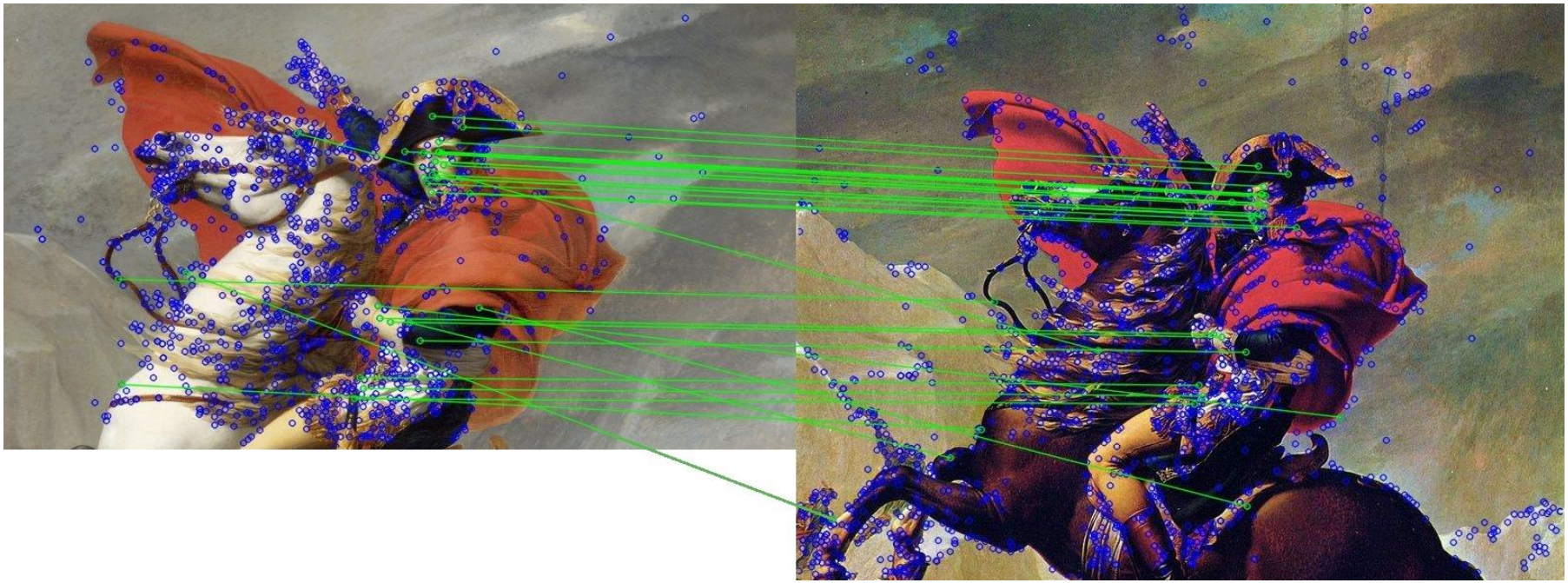
- FAST to find key points
- BRIEF to represent descriptors

FLANN: Fast Library for Approximate Nearest Neighbors

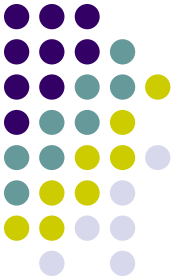
- It contains a collection of algorithms optimized for fast NN search
- Suitable for large datasets and for high dimensional features

Feature matching

Cosine similarity = 0.4887



Number of matching points = 38



Experiments

Experiments

Base network: Modified pre-trained VGG16

Training set:
Original image

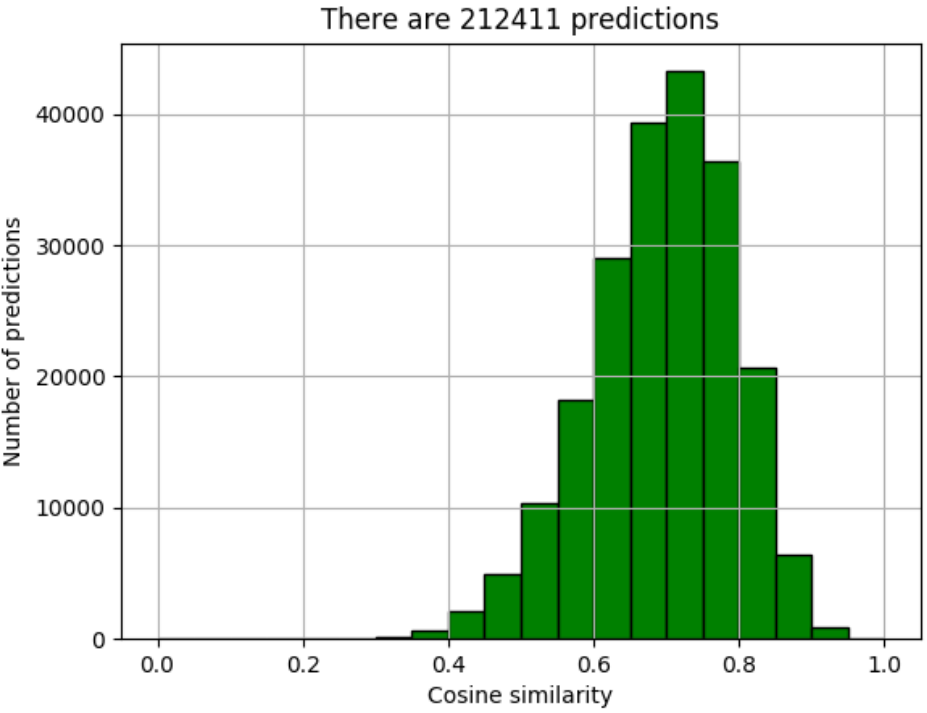
Test set:
Augmented images



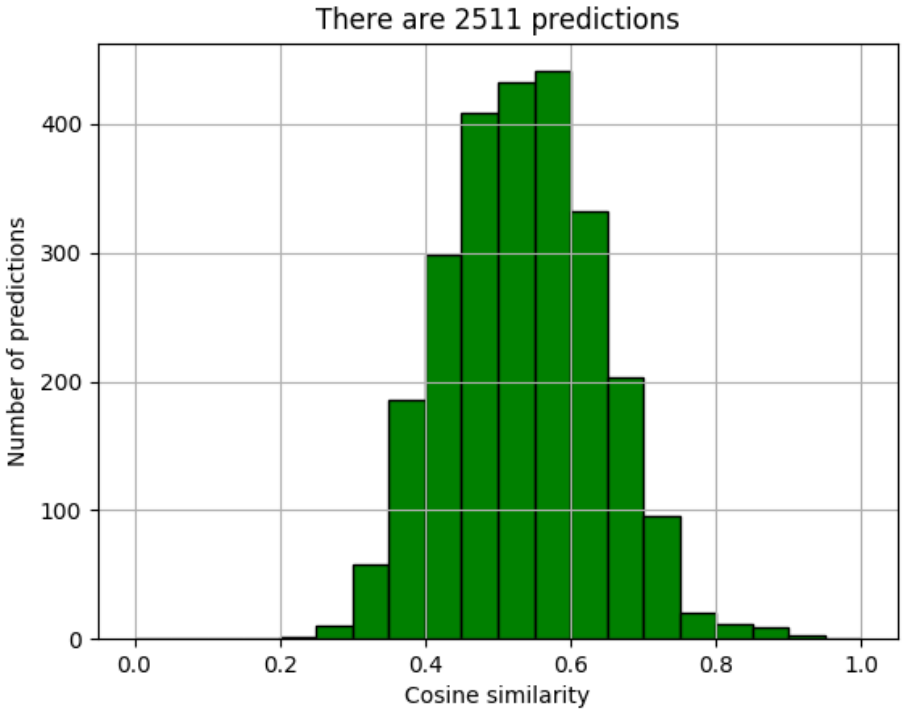



Statistics

Correct predictions



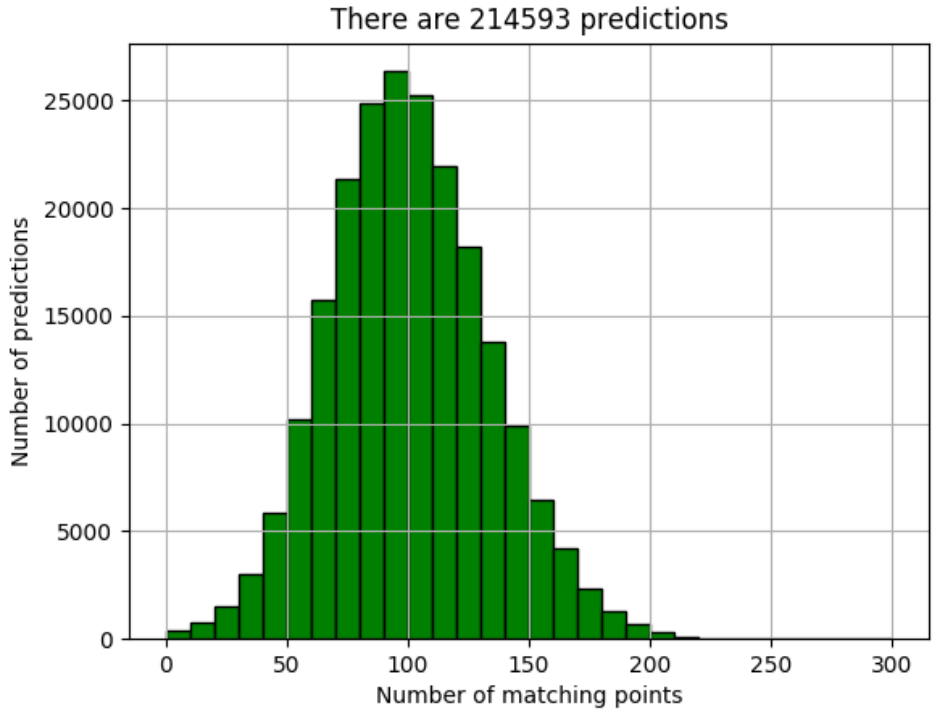
Wrong predictions



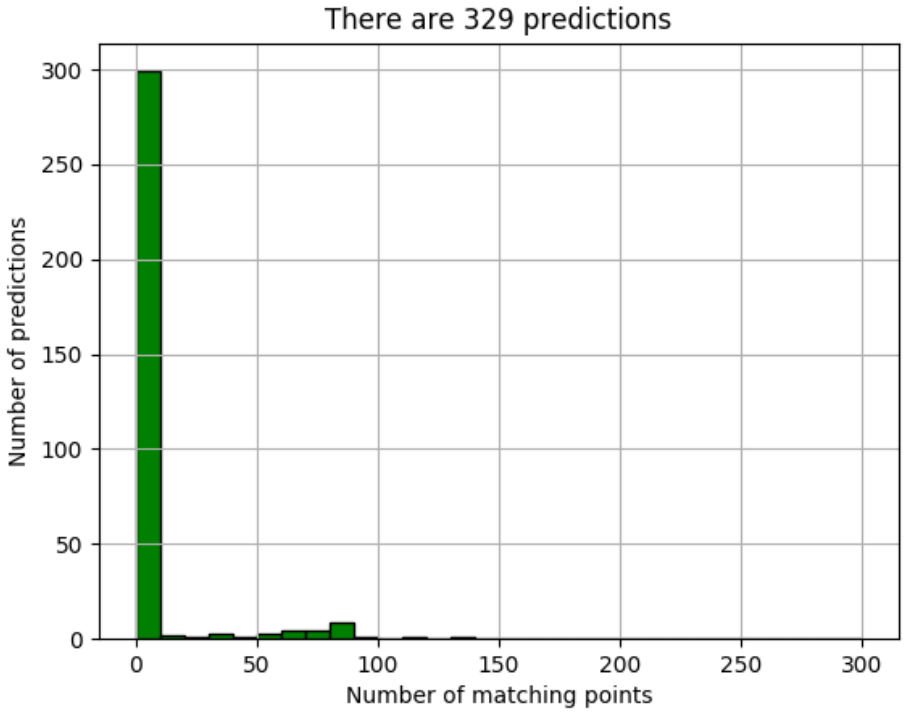
Post-processing statistics

We consider only the top *N* (*N*=100) images according to their cosine similarity

Correct predictions



Wrong predictions



Duplicated images in the dataset

Query image	Predicted image	Correct image
		
		
		

Search algorithm

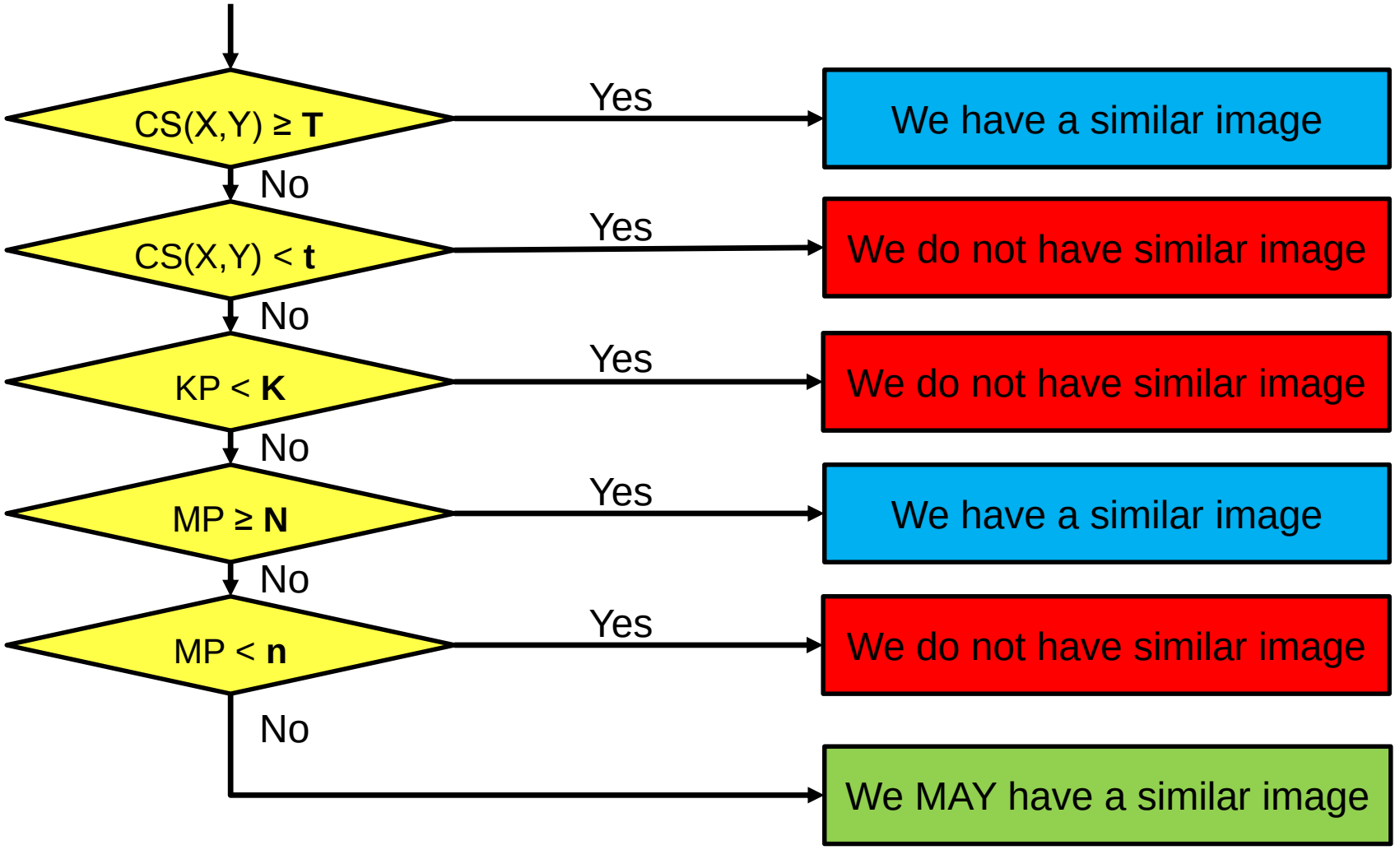
- ***T*** and ***t*** are the upper and lower thresholds, resp., for cosine similarity
- ***K*** is the minimum number of required key points
- ***N*** and ***n*** are the upper and lower thresholds, resp., for matching points

Best thresholds

Parameter	Threshold
Cosine similarity	(T, t) = (0.91, 0.1)
Minimum key points	10
Number of matching points	(N, n) = (20, 10)

Recall: 214239/214269 (**99.986%)**
False negative: **653**
Precision: **100%**

Search algorithm

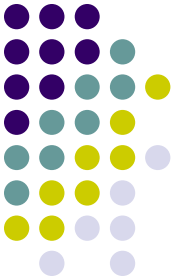


Conclusion

- We presented a robust model, which achieves high accuracy
- The run time is ~1.8 sec per image
- *Google Images* can do the same task, but less robust against geometric deformations

Future work:

- Fine-tune VGG network with specific dataset
- Consider faster post-processing



Demo

Questions?