# Class-Weighted Convolutional Features for Image Retrieval



Albert Jiménez


Xavier Giró-i-Nieto


Jose Alvarez

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

DATA 61 CSIRO

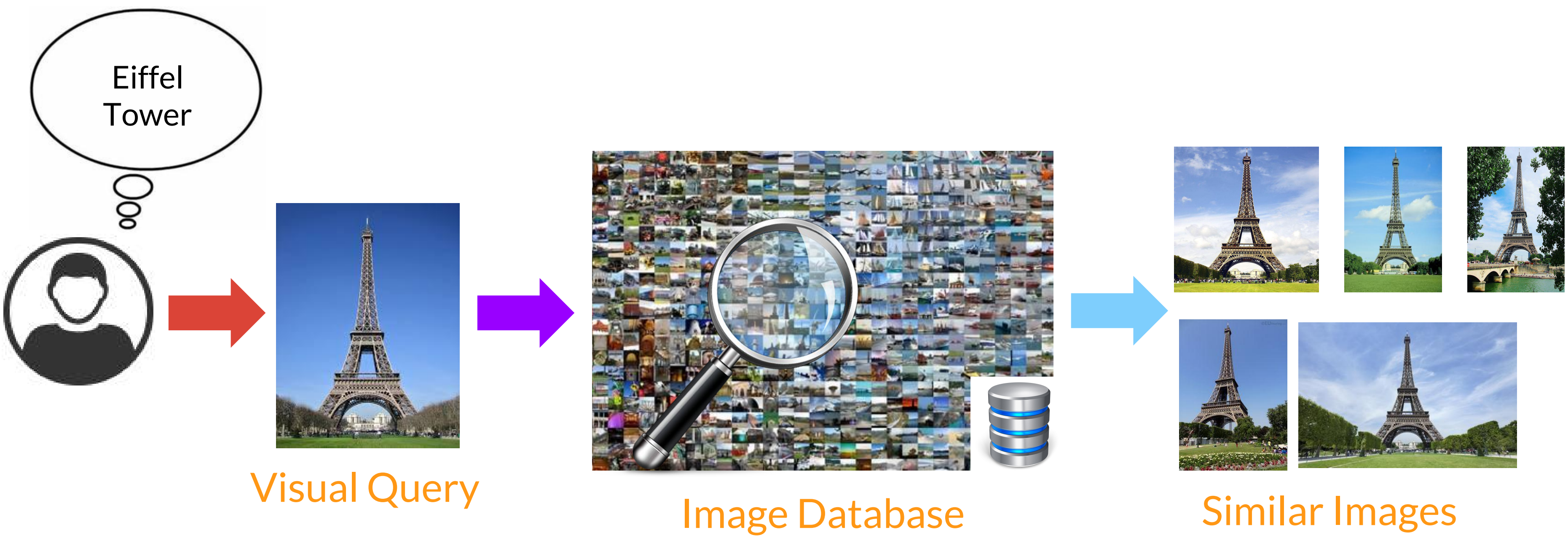

Code available at GitHub

# Outline

- ▷ Introduction
- ▷ Related Work
- ▷ Our Proposal
- ▷ Experiments
- ▷ Conclusions

# 1. Introduction

# Visual Instance Retrieval

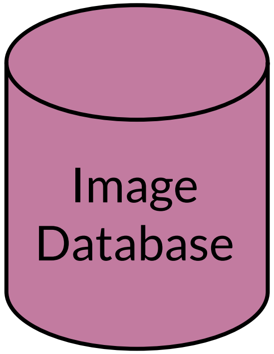



Given an image query, generate a ranked list of similar images

# Visual Instance Retrieval

Query image | Image Representations | Image Matching | Ranking List

$v = (v_1, \ldots, v_n)$

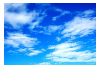



$v_1 = (v_{11}, \ldots, v_{1n})$

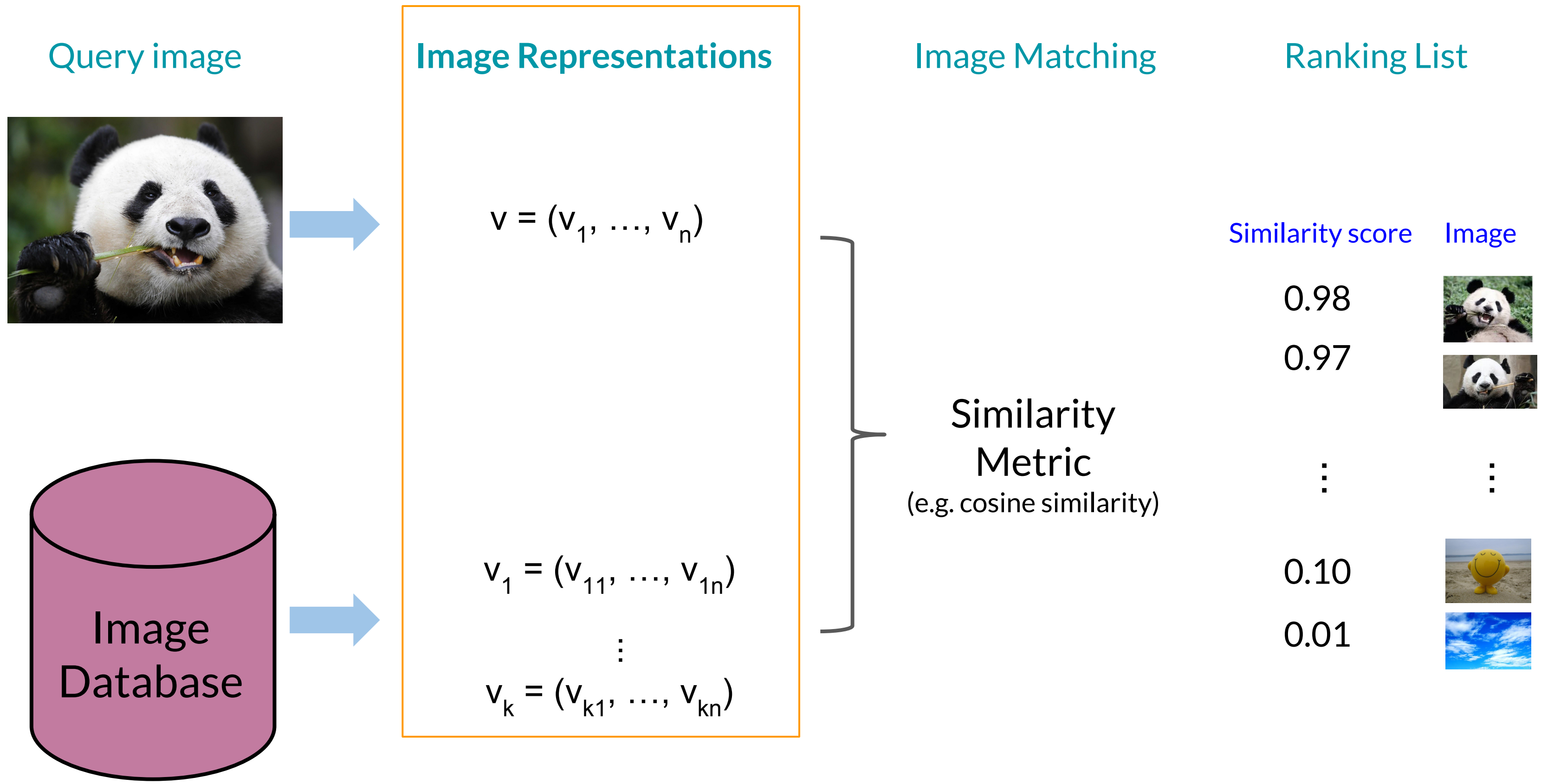
$\vdots$

$v_k = (v_{k1}, \ldots, v_{kn})$

$\Bigg\}$ Similarity Metric
(e.g. cosine similarity)

| Similarity score | Image |
|---|---|
| 0.98 |  |
| 0.97 |  |
| ⋮ | ⋮ |
| 0.10 |  |
| 0.01 |  |

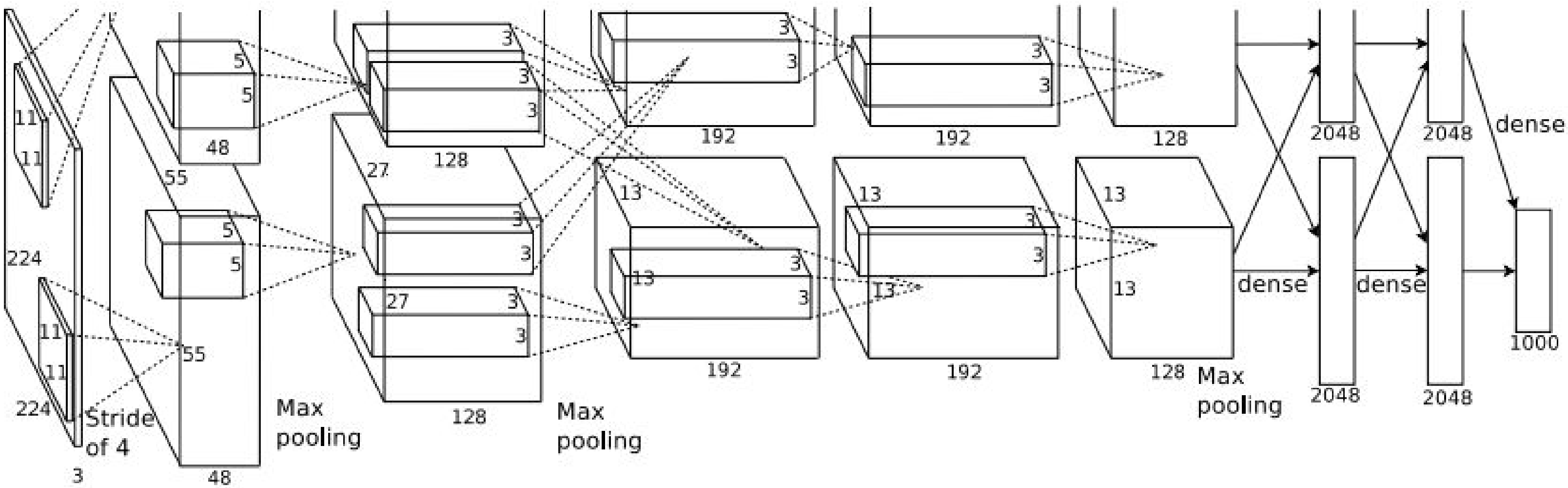
# Visual Instance Retrieval



# 2. Related Work

# Image Representations

## Convolutional Neural Networks



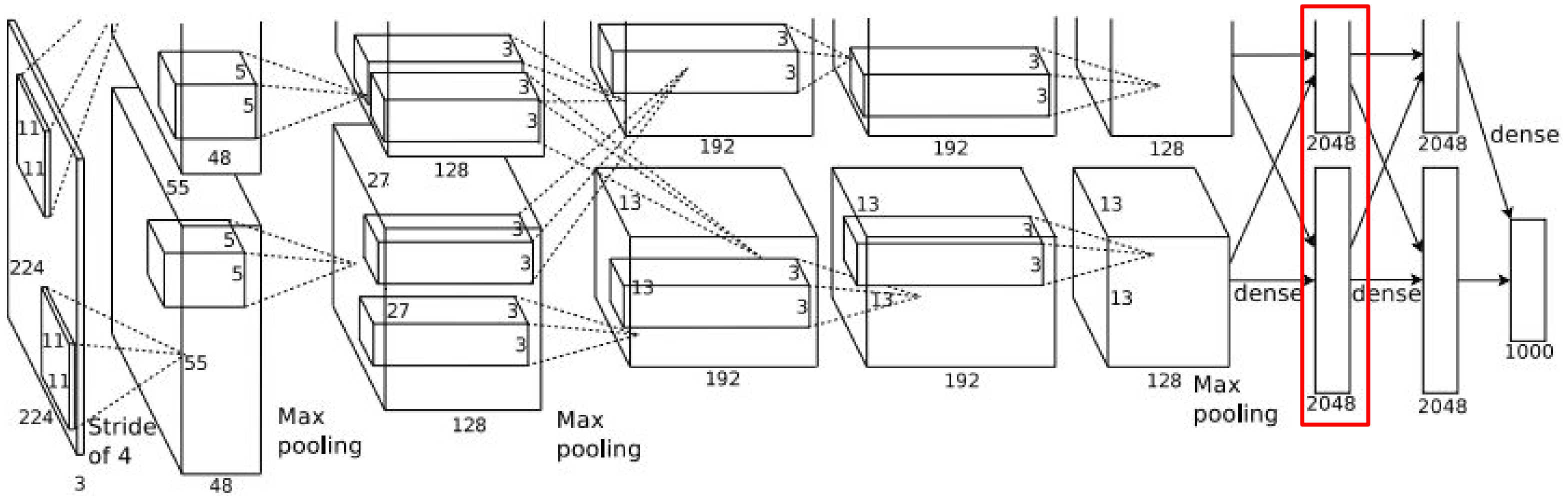

Example of a CNN: AlexNet


Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

# Image Representations

## Convolutional Neural Networks

**Fully-Connected features** as global representation



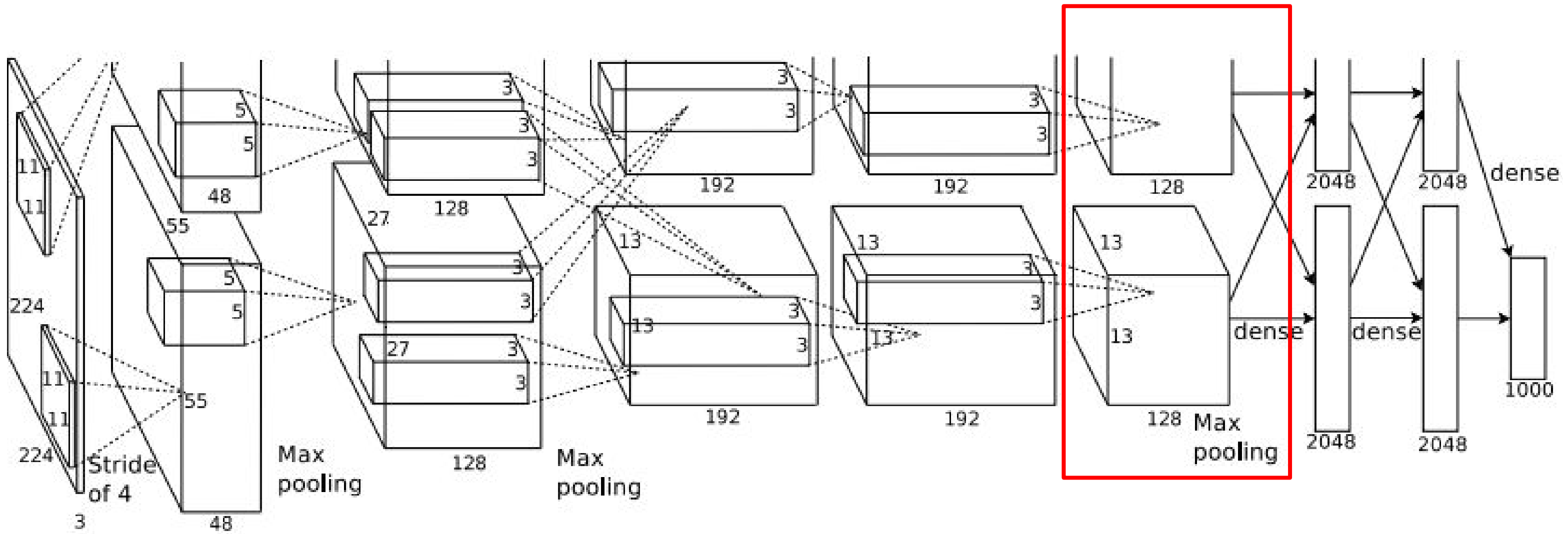


Babenko, A., Slesarev, A., Chigorin, A., & Lempitsky, V. (2014). Neural codes for image retrieval. In *ECCV 2014*
Razavian, A., Azizpour, H., Sullivan, J., & Carlsson, S. (2014). CNN features off-the-shelf: an astounding baseline for recognition. In DeepVision *CVPRW 2014*

# Image Representations

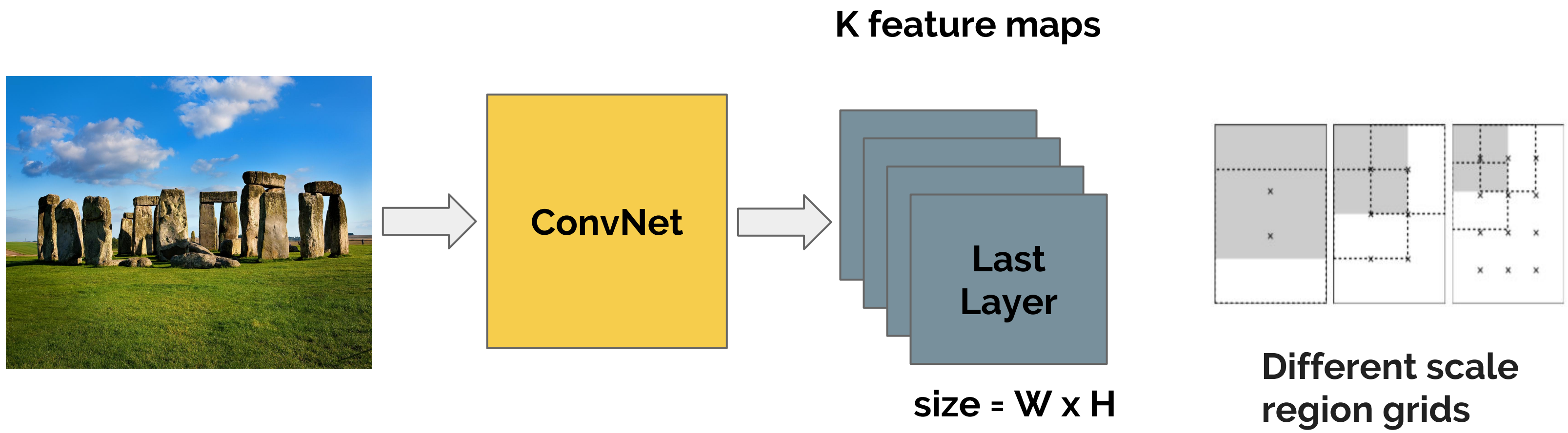
## Convolutional Neural Networks

**Convolutional features** (Sum/Max Pooled) as global representations




Babenko, A., & Lempitsky, V. (2015). Aggregating local deep features for image retrieval. *ICCV 2015*
Tolias, G., Sicre, R., & Jégou, H. (2015). Particular object retrieval with integral max-pooling of CNN activations. *ICLR 2016*
Kalantidis, Y., Mellina, C., & Osindero, S. (2015). Cross-dimensional Weighting for Aggregated Deep Convolutional Features. *arXiv preprint arXiv:1512.04065*.

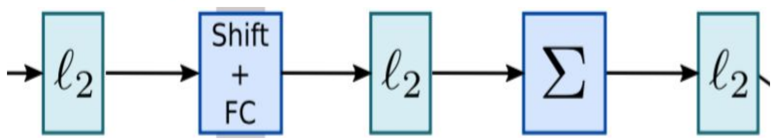
# R-MAC




- Regions selected using a rigid grid
- Compute a feature vector per region
- Combine all region feature vectors
    - Dimension → 256 / 512
    - AlexNet / VGG-16

maximum activation

$$\mathbf{f}_{\mathcal{R}} = [\mathrm{f}_{\mathcal{R},1} \ldots \boxed{\mathrm{f}_{\mathcal{R},i}} \ldots \mathrm{f}_{\mathcal{R},K}]^{\top}$$

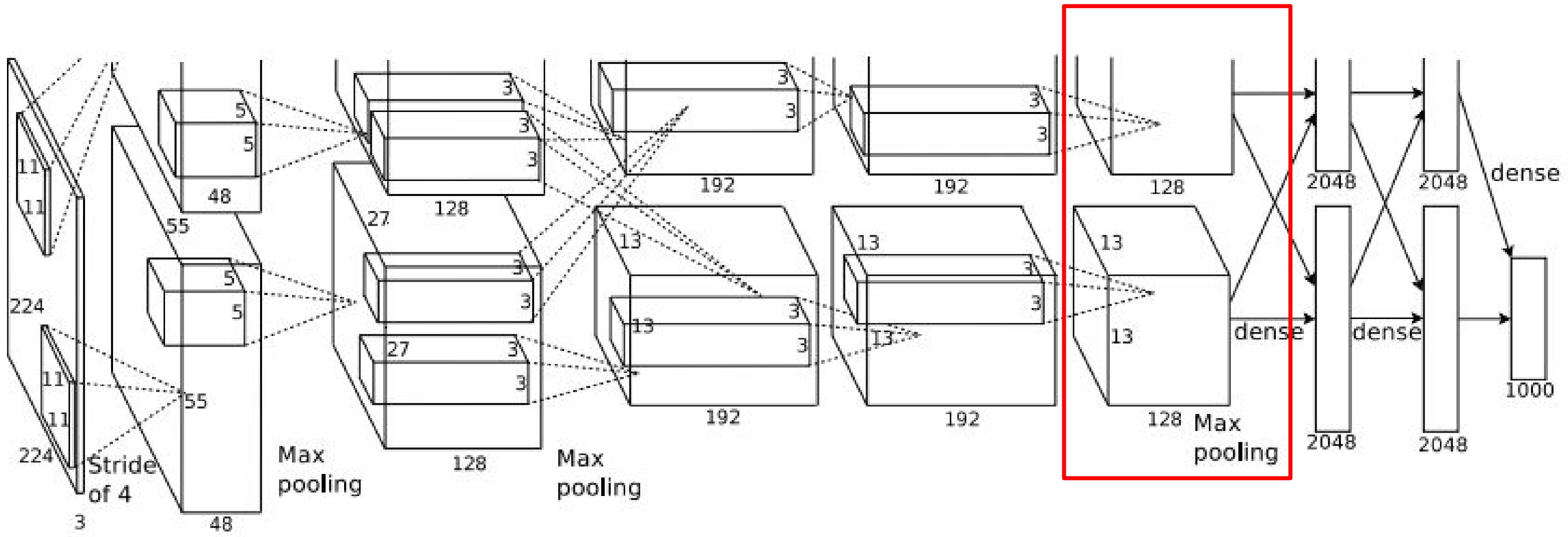



Tolias, G., Sicre, R., & Jégou, H. (2015). Particular object retrieval with integral max-pooling of CNN activations. *ICLR 2015*

# Image Representations

## Convolutional Neural Networks

**Convolutional features** - Encoded with VLAD or BoW

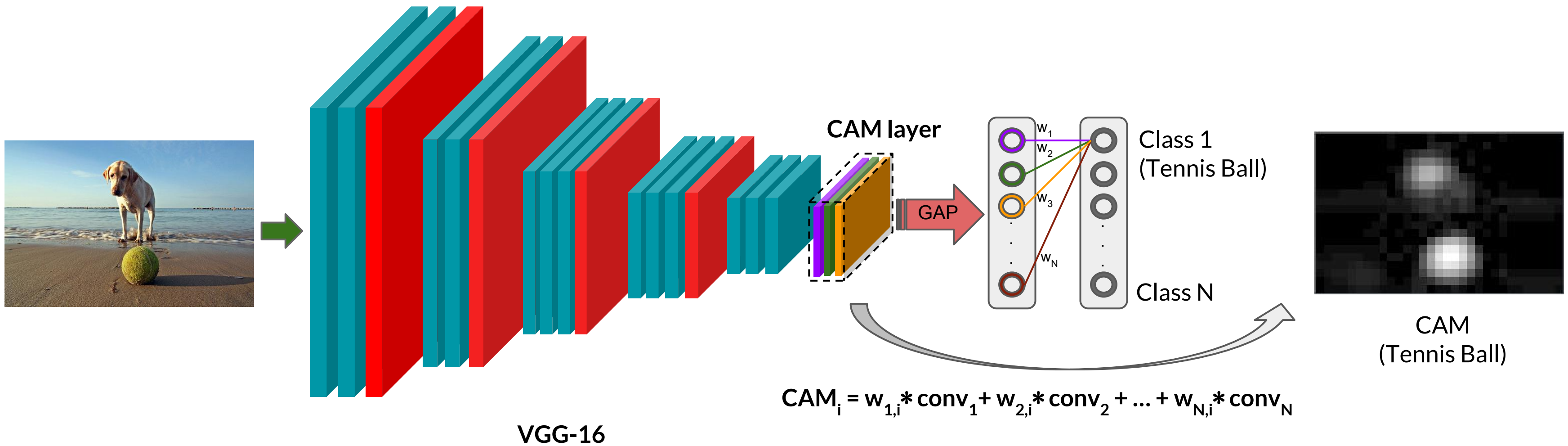



Ng, J., Yang, F., & Davis, L. (2015). Exploiting local features from deep networks for image retrieval. In *DeepVision CVPRW 2015*
E. Mohedano, A. Salvador, K. McGuinness, F. Marques, N. E. O'Connor and X. Giro, Bags of Local Convolutional Features for Scalable Instance Search. In ICMR 2016

# Motivation: Image Semantics

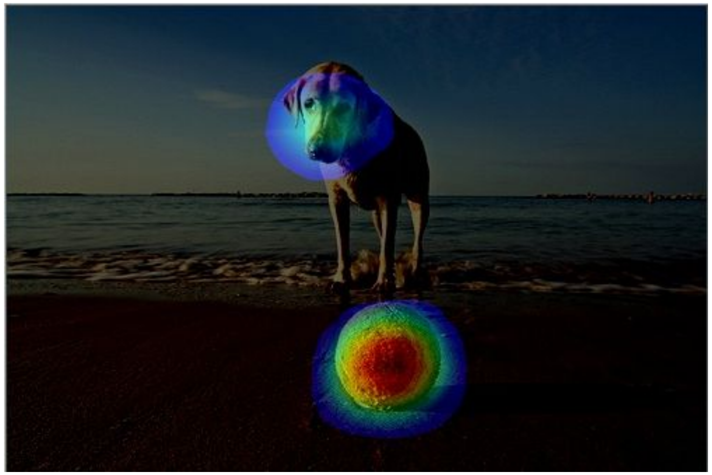
# Motivation: Image Semantics

## Class Activation Maps




B. Zhou, A. Khosla, Lapedriza. A., A. Oliva, and A. Torralba. 2016. Learning Deep Features for Discriminative Localization. CVPR (2016).

# Motivation: Image Semantics



**Tennis Ball**

**Chesapeake Bay Retriever**

**Simple Classes**

# Motivation: Image Semantics

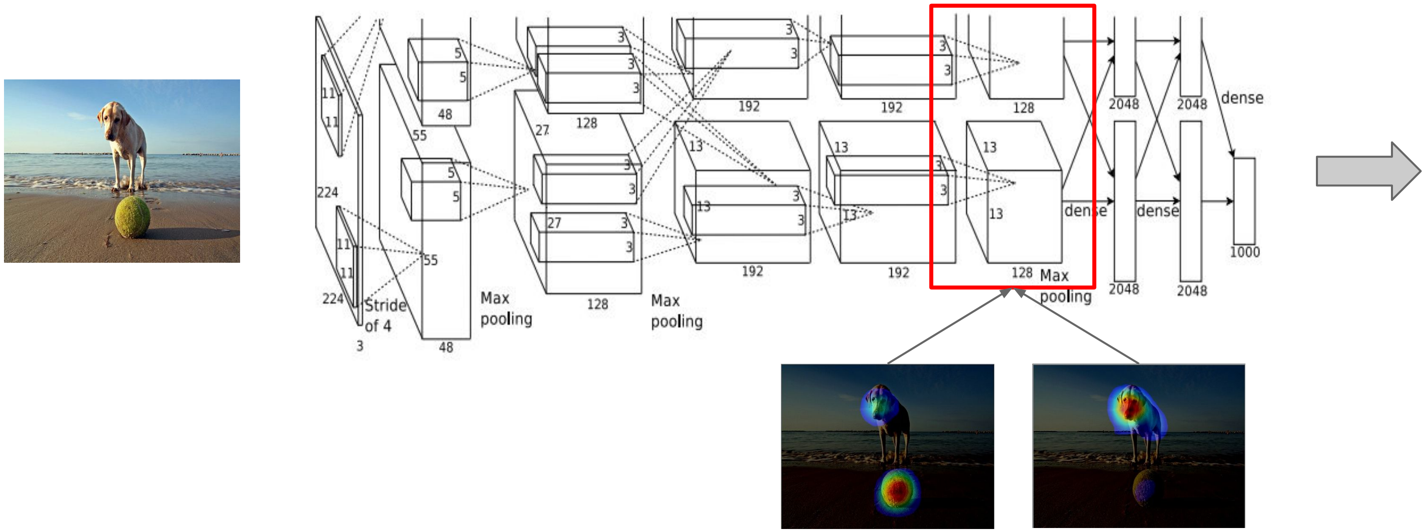
**Sand Bar**

**Seashore**

**Complex Classes**

# 3. Proposal

# Proposal



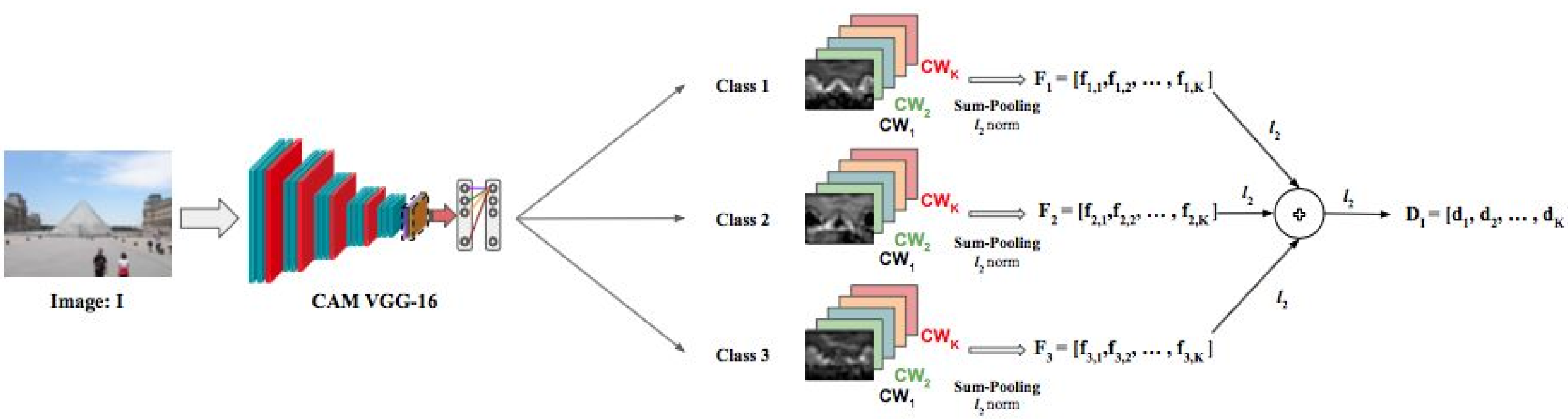

**Compact Descriptor**

**Class-Weighted Convolutional Features**

Encode images combining **image semantics** knowledge with **convolutional features**

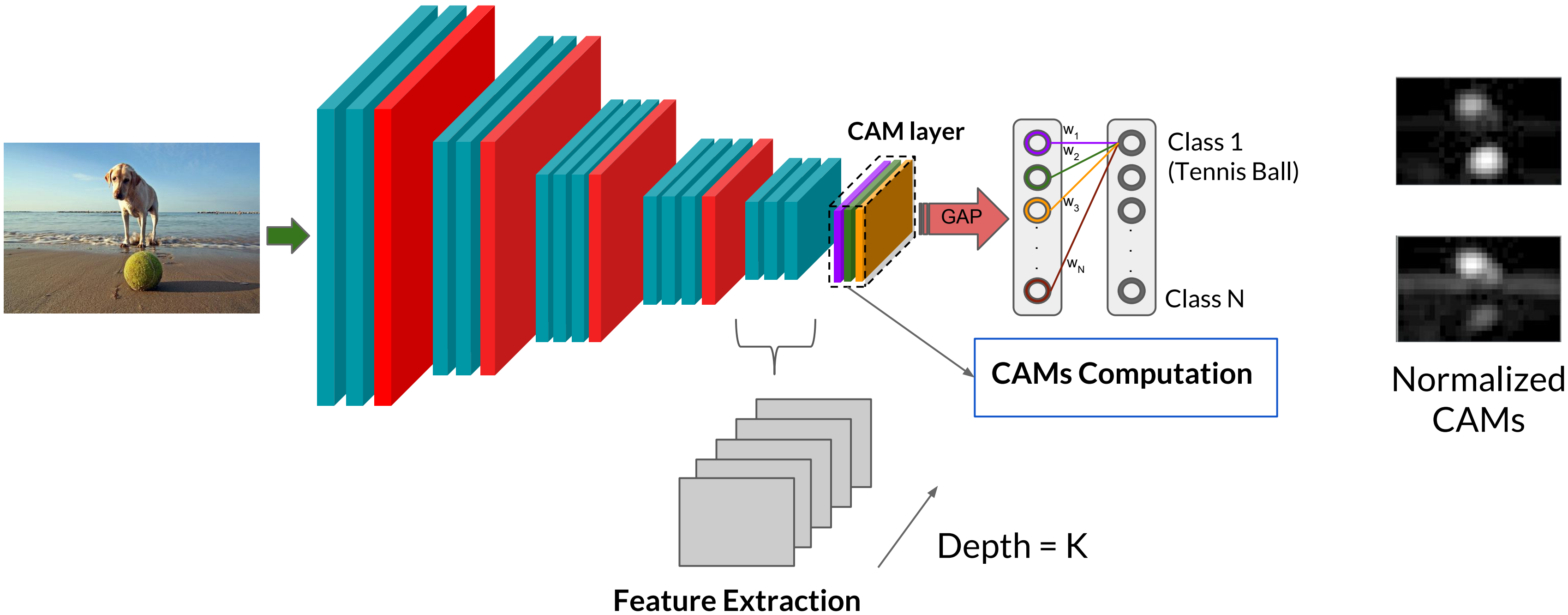
# Retrieval Pipeline

1. Convolutional Features & CAMs Extraction
2. Channel & Class-Weighting
3. Feature Pooling
4. Descriptor Aggregation



# Retrieval Pipeline

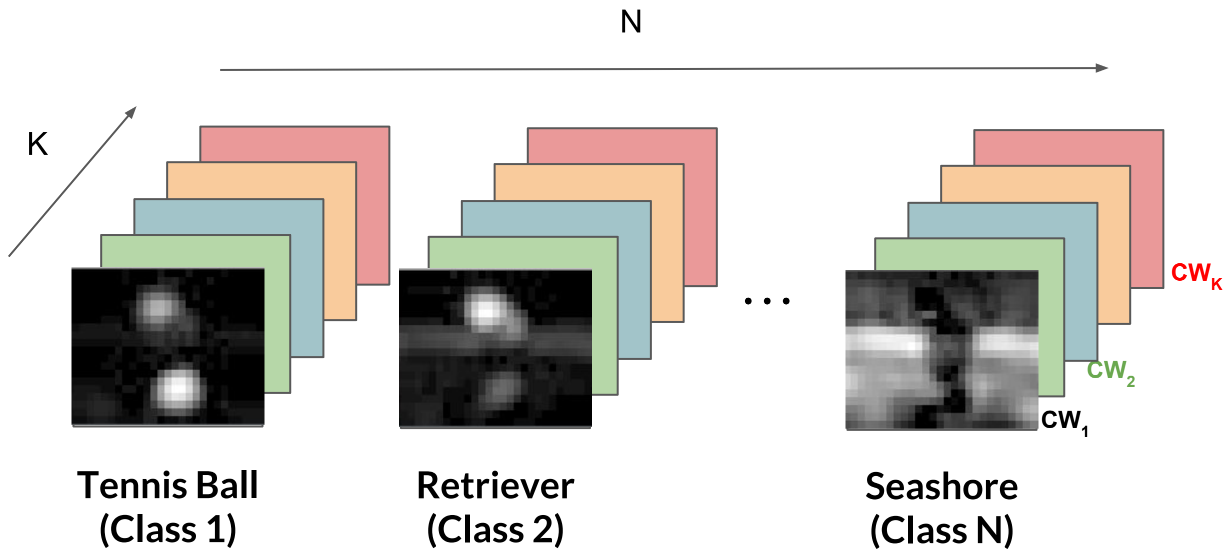
1. Convolutional Features & CAMs Extraction




In a **single forward pass** we extract convolutional features and image CAMs

# Retrieval Pipeline

2. Channel & Class-Spatial Weighting

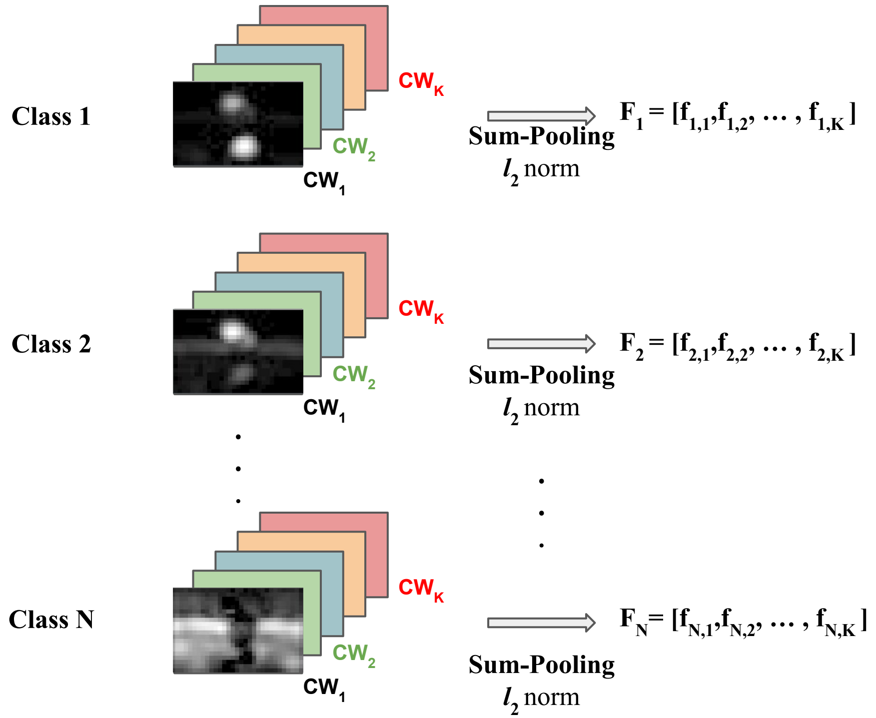



**Spatial Weighting** based on Class Activation Maps

**Channel Weighting** based on feature maps sparsity (as in CroW)

# Retrieval Pipeline

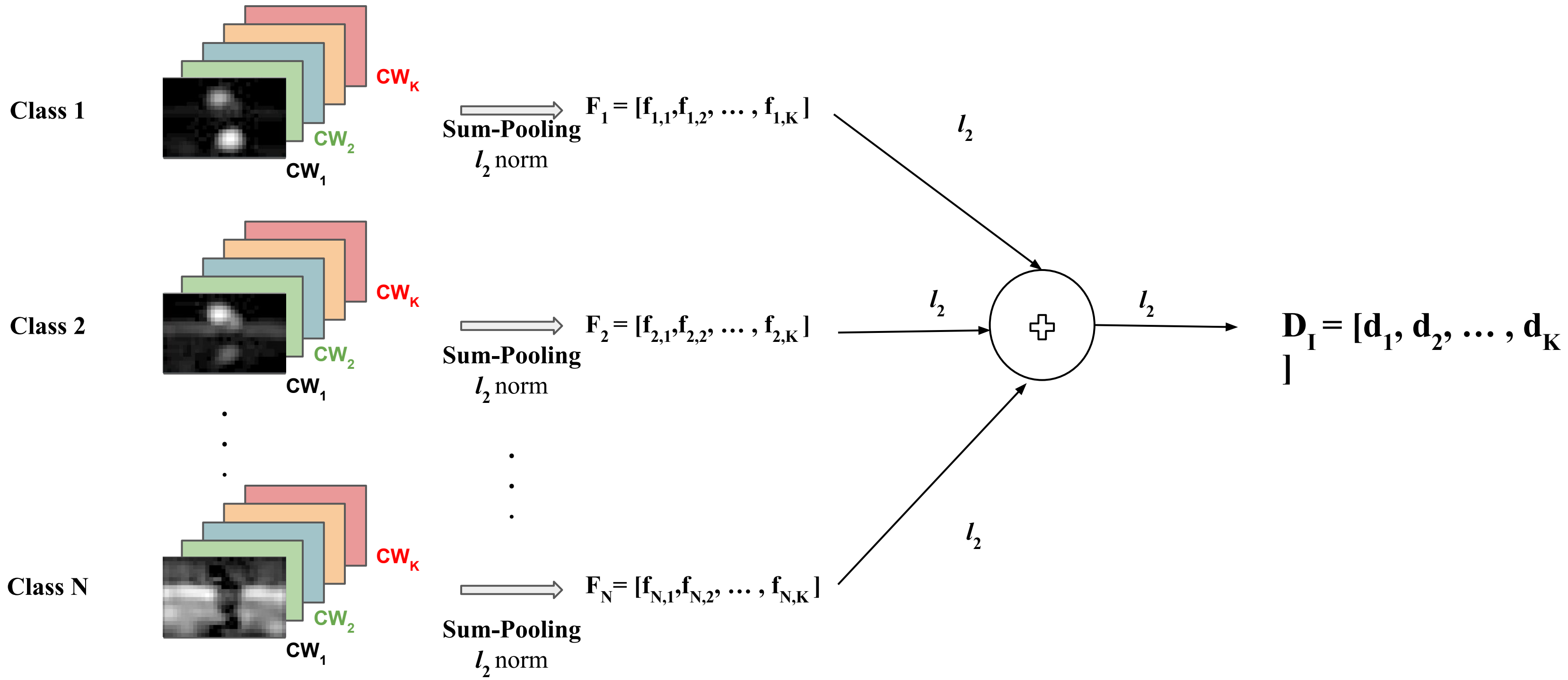
3. Feature Pooling




One vector per class: $F_c = [f_{c,1}, f_{c,2}, \dots, f_{c,K}]\ \forall c \in [1, N]$

# Retrieval Pipeline

## 4. Descriptor Aggregation



# Descriptor Aggregation Strategies

## Online Aggregation (OnA)

Image Dataset

Our Retrieval Pipeline

Store all the class vectors ($F_c$)

Query

Our Retrieval Pipeline

Query Descriptor

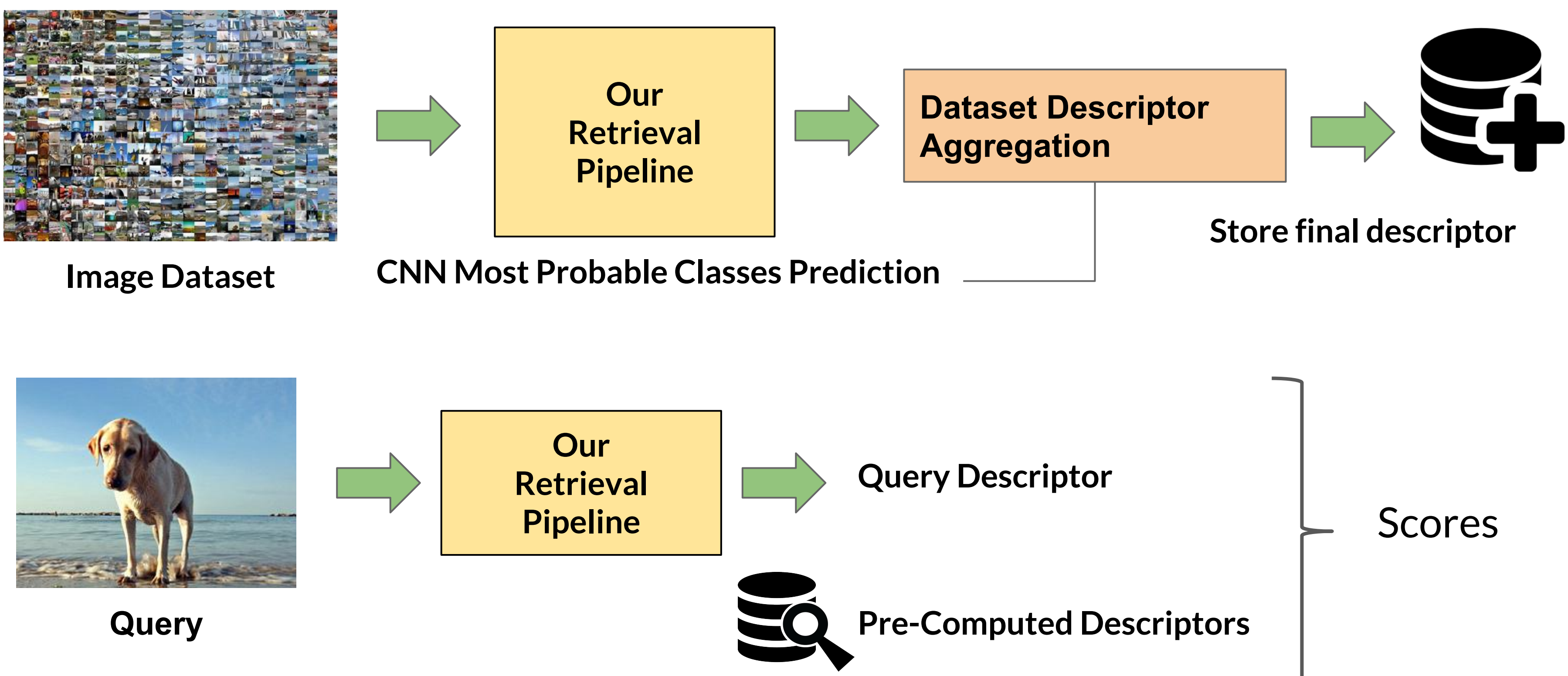
Dataset Descriptor Aggregation

Scores

CNN Most Probable Classes Prediction from query (Retriever, seashore...)

# Descriptor Aggregation Strategies

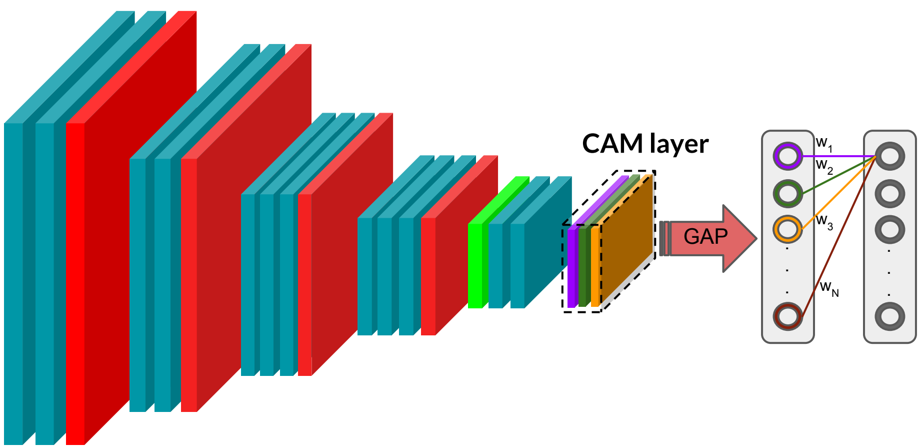
## Offline Aggregation (OfA)



# 4. Experiments

# Experimental Setup

- Keras over Theano
- Images resized to 1024x720 (keeping aspect ratio)
- VGG-16 CAM model as feature extractor (ImageNet)
- Features from *Conv5_1* Layer

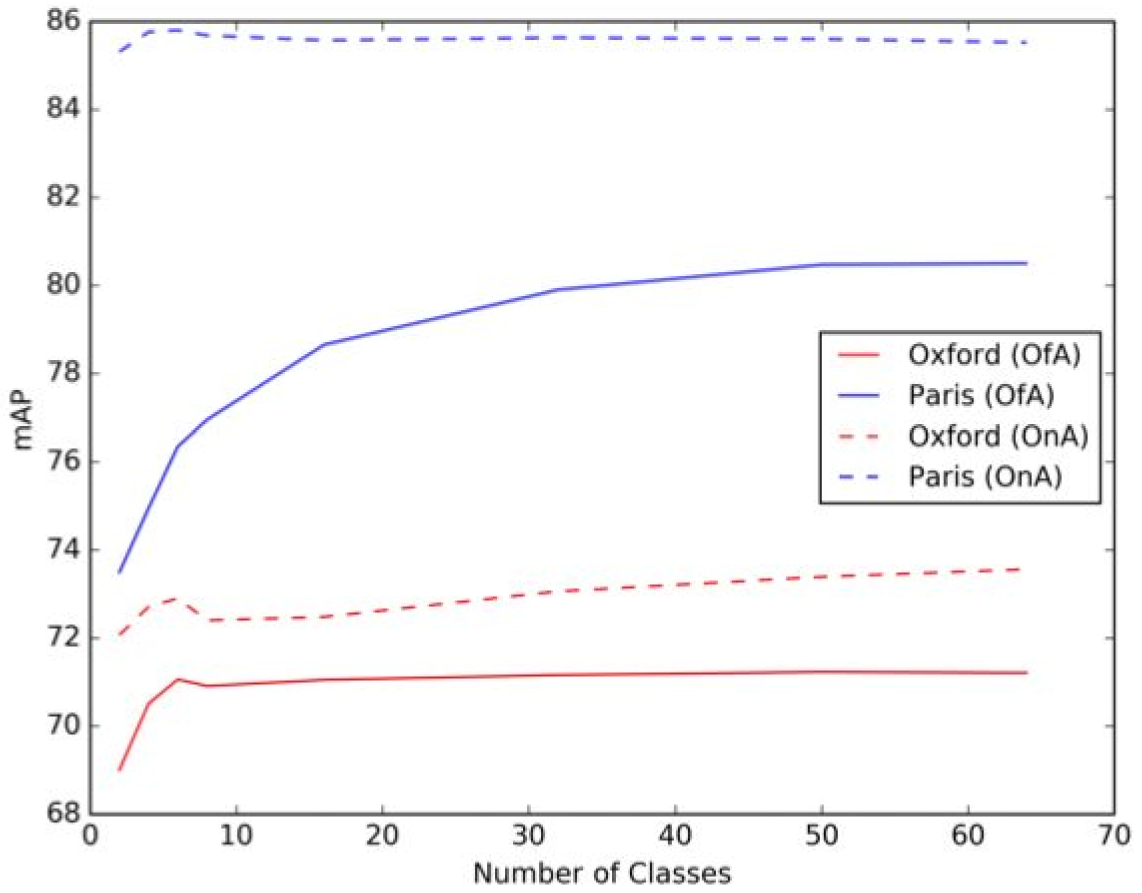


# Experimental Setup

- Datasets
  - Oxford 5k
  - Paris 6k
  - 100k Distractors (Flickr)
- Scores computed with cosine similarity
- Evaluation metric:  mean Average Precision (mAP)

Philbin, J. , Chum, O. , Isard, M. , Sivic, J. and Zisserman, A. Object retrieval with large vocabularies and fast spatial matching,  *CVPR 2007*
Philbin, J. , Chum, O. , Isard, M. , Sivic, J. and Zisserman, A. Lost in Quantization: Improving Particular Object Retrieval in Large Scale Image Databases. *CVPR 2008*

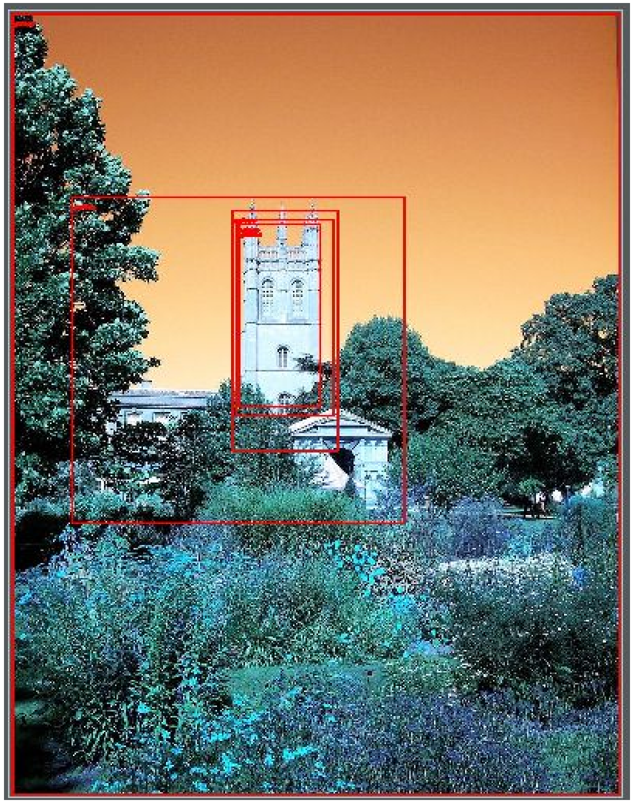
# How many objects are relevant?

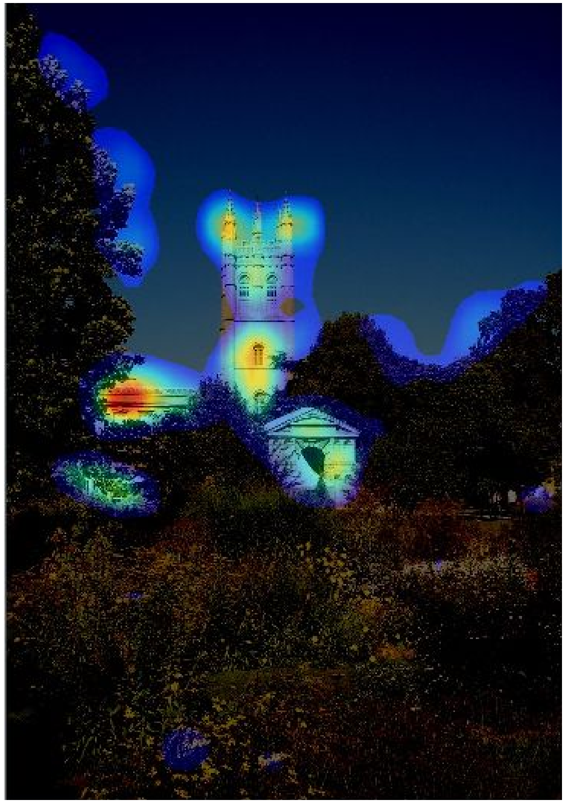


# Comparison with State-of-the-Art

| Method | Dim | Oxf5k | Par6k | Oxf105k | Par106k |
|---|---|---|---|---|---|
| SPoC[2] | 256 | 0.531 | - | 0.501 | - |
| CroW[10] | 512 | 0.682 | 0.796 | 0.632 | 0.710 |
| uCroW[10] | 256 | 0.666 | 0.767 | 0.629 | 0,695 |
| Razavian [19] | 32k | **0.843** | 0.853 | - | - |
| R-MAC[27] | 512 | 0.669 | 0.830 | 0.616 | **0.757** |
| BoW[12] | 25k | 0.738 | 0.820 | 0.593 | 0.648 |
| **Ours(OnA)** | 512 | 0.729 | **0.858** | - | - |
| **Ours(OfA)** | 512 | 0.712 | 0.799 | **0.672** | 0.727 |

# Re-Ranking





# Comparison with State-of-the-Art

| Method | Dim | R | QE | Oxf5k | Par6k | Oxf105k | Par106k |
|---|---|---|---|---|---|---|---|
| CroW | 512 | - | 10 | 0.722 | 0.855 | 0.678 | 0.797 |
| Ours(OnA) | 512 | - | 10 | 0.766 | **0.879** | - | - |
| Ours(OfA) | 512 | - | 10 | 0.737 | 0.835 | 0.714 | 0.777 |
| BoW | 25k | 100 | 10 | 0.788 | 0.848 | 0.651 | 0.641 |
| Ours(OnA) | 512 | 100 | 10 | 0.786 | 0.876 | - | - |
| Ours(OfA) | 512 | 100 | 10 | 0.772 | 0.836 | 0.744 | 0.777 |
| RMAC | 512 | 1000 | 5 | 0.770 | 0.877 | 0.726 | **0.817** |
| Ours(OnA) | 512 | 1000 | 5 | **0.812** | 0.874 | - | - |
| Ours(OfA) | 512 | 1000 | 5 | 0.803 | 0.854 | **0.765** | 0.778 |

# 5. Conclusions

# Conclusions

- We have proposed to use the semantic information of images to encode them.
- We introduced the use of CAMs to spatially weight convolutional features inside a retrieval pipeline.
- We demonstrated that our retrieval system outperforms the previous state-of-the-art in off-the-shelf image retrieval.


Thank you