THE INTELLIGENT IMAGE

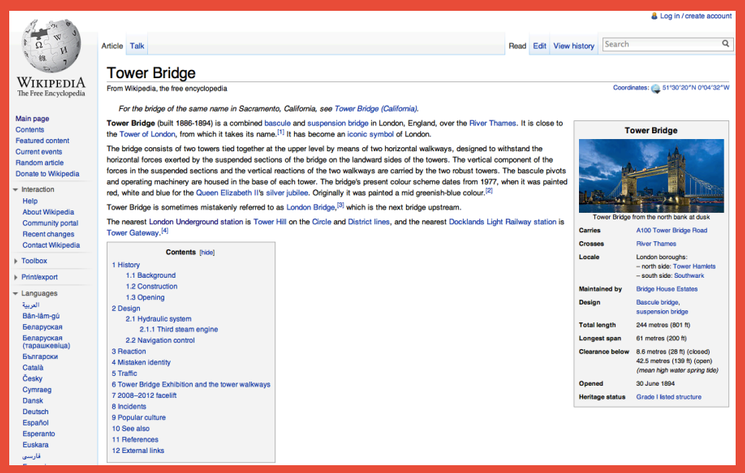
LARGE SCALE OBJECT RECOGNITION

THE GOAL:
AUTOMATICALLY RECOGNISE AND TAG OBJECTS IN IMAGES.

IMPLEMENTATION:
A WEBSITE WHERE USERS CAN UPLOAD A PHOTO TO BE MADE IN TO AN INTELLIGENT IMAGE.

NORMAL IMAGE

INTELLIGENT IMAGE



EACH ARTICLE ON WIKIPEDIA DEFINES AN *OBJECT*. ALL THE IMAGES FROM AN OBJECT'S ARTICLE ARE DOWNLOADED TO A DATABASE AND DEFINE THE MODEL OF THAT OBJECT

MAX JADERBERG
SUPERVISOR: ANDREW ZISSERMAN

HOW IT WORKS:

1. EXTRACT VISUAL WORDS

SCALE **I**NVARIANT **F**EATURE **T**RANSFORM

$$\begin{pmatrix} x_1 & \cdots & x_N \\ y_1 & \cdots & y_N \\ s_1 & \cdots & s_N \\ \theta_1 & \cdots & \theta_N \end{pmatrix}_{4\times N} \qquad (v_1 | \ldots \ldots \ldots | v_N)_{128\times N}$$

The features of an image are detected and described by a 128-dimensional vector. The feature space is then quantized, creating "visual words".

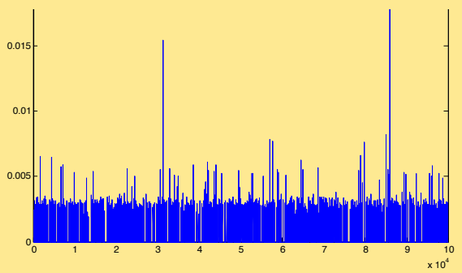
FEATURES

↓

WORDS

↓

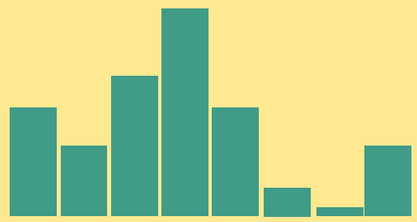
HISTOGRAM

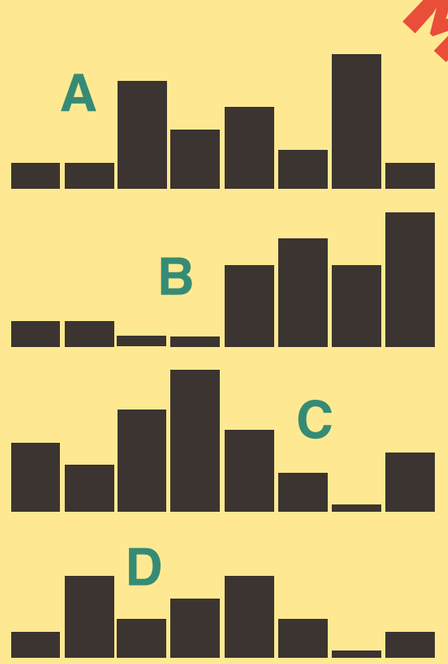


2. GOOGLE STYLE SEARCH

```
matches = search(Database for Words);
```

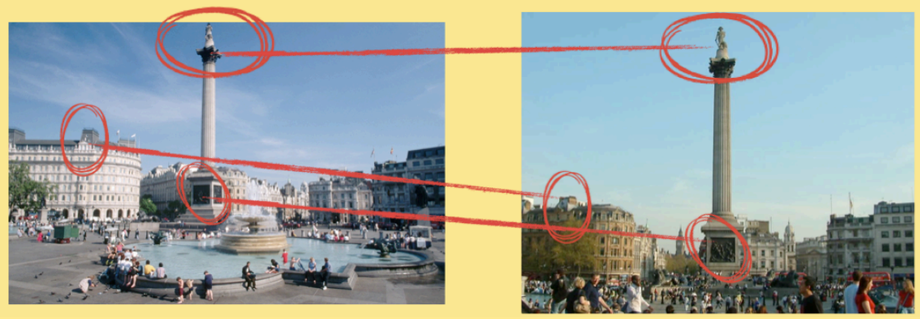
The tf-idf weighted histograms are then used to search the database for similar histograms. This is the same method as Google uses for text search, replacing words with "visual words".





Most similar!

3. SPATIALLY VERIFY



$$X_{db} = \begin{bmatrix} A & \vec{t} \\ 0 & 1 \end{bmatrix} X_{query}$$

The match from the database and query image must depict the same object, so there should be a spatial correspondence between the visual words of the two images.

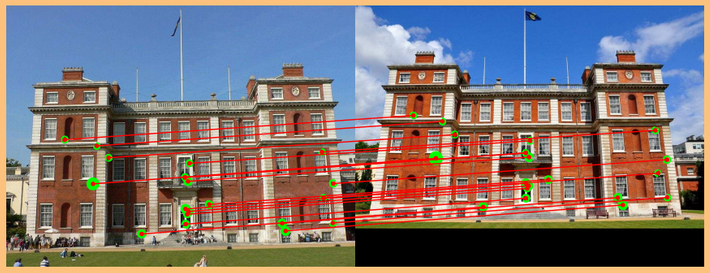
IMPROVEMENTS:

RANSAC

inliers: 6

Initially RANSAC was used to estimate the affine transformation relating the visual words of two features.

inliers: 20



NOSAC

This was replaced by a method called NOSAC which uses the scale of each visual word correspondence to improve the estimation of the transformation.

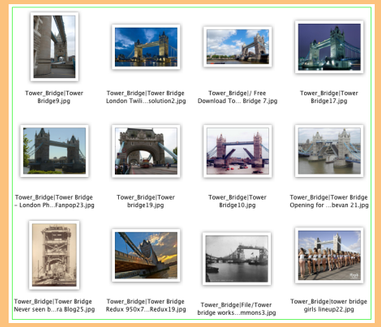
AFFINE INVARIANT

An affine invariant feature detector and descriptor was used. This means that features that are skewed due to changes in viewpoint can still be matched together. Each feature is now represented by an ellipse, with a pair of features being able to estimate a full affine transformation.

TURBO-BOOSTING

A novel method was developed to increase the information content of the database of Wikipedia images (called *model images*) by augmenting them with the visual words of images from Microsoft Bing (called *turbo images*).

(a) Model images



(b) Turbo images

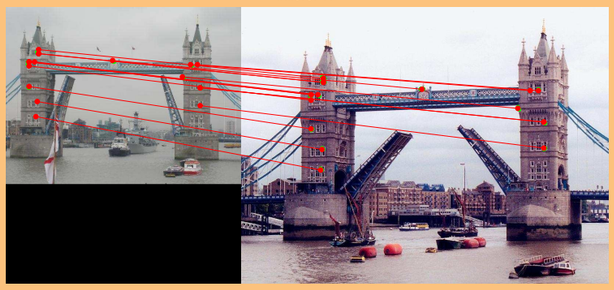
1. DOWNLOAD IMAGES FROM BING FOR EACH OBJECT

2. COMPARE EACH MODEL IMAGE WITH EACH TURBO IMAGE

(c) Model image words

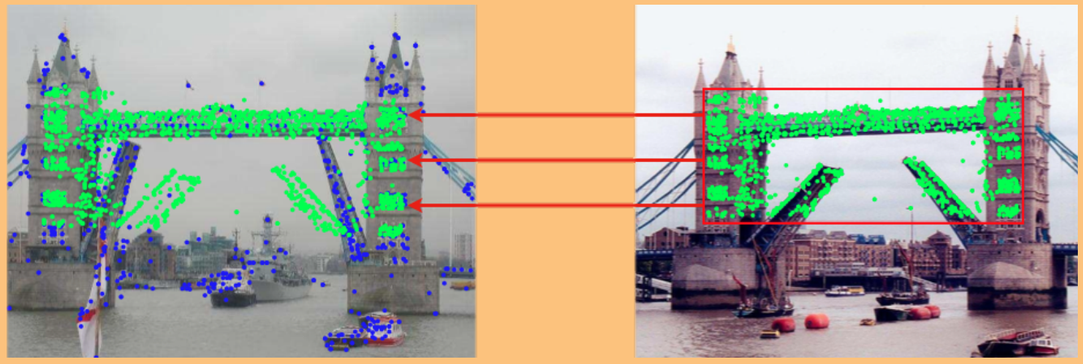
(d) Turbo image words

inliers: 17



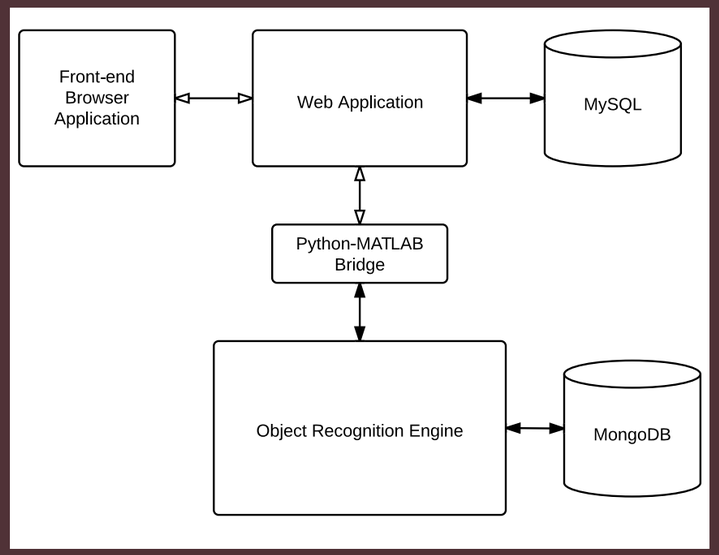
(e) The turbo image is spatially verified with the model image

3. IF A MATCH, PROJECT THE WORDS FROM THE TURBO IMAGE ON TO THE MODEL IMAGE



(f) The words in the spatially verified region are used to augment the model image

IMPLEMENTATION



THE OBJECT RECOGNITION AND TAGGING ENGINE WAS BUILT IN MATLAB. THIS IS CONNECTED TO A WEBSERVER WRITTEN IN PYTHON VIA A MATLAB-PYTHON BRIDGE. THE WEBSERVER DELIVERS AN HTML AND JAVASCRIPT APP TO THE BROWSER.

RESULTS

BASELINE SYSTEM:	**18.1%**
+ NOSAC:	**20.8%**
+ AFFINE INVARIANT:	**22.5%**
+ ROOTSIF:	**24.5%**
+ TURBO-BOOSTING:	**31.7%**

Performance measured by *yield* - the percentage of test images successfully recognized.