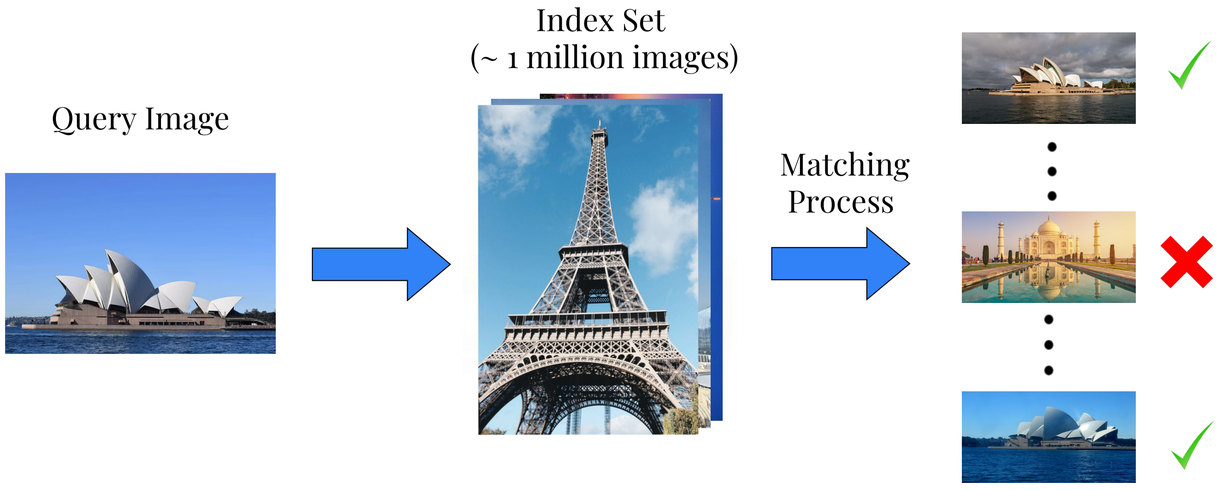
Google Landmarks Recognition and Retrieval Challenge

Maxence Dutreix, Nathan Hatch

Raghav Kuppan, Pranav S, Anirudha Sundaresan

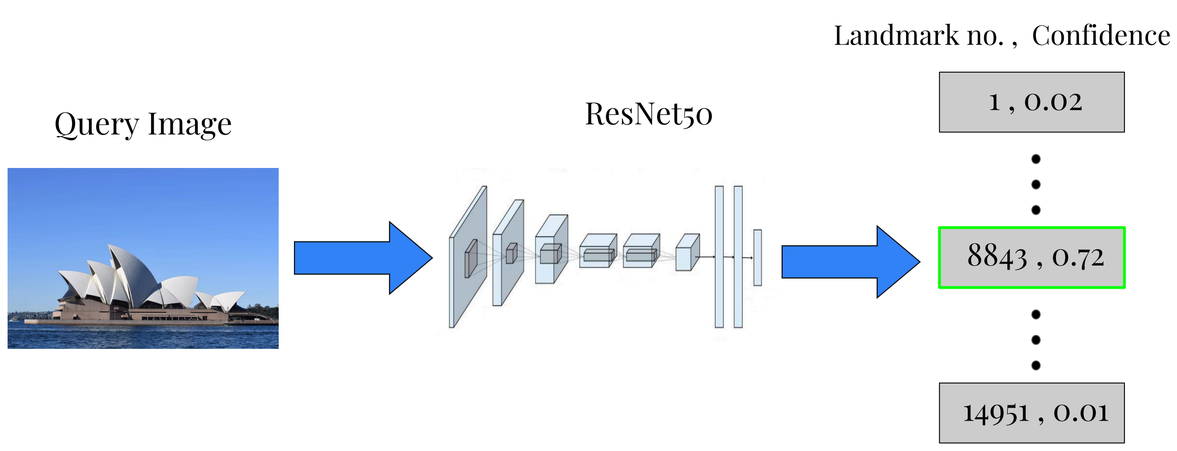
Did you ever go through your vacation photos and ask yourself: What is the name of this temple I visited in China? Who created this monument I saw in France? Or try to find more pictures of you with Taj Mahal? Landmark recognition and retrieval can help! This project can predict landmark labels directly from image pixels and also retrieve similar images in your collection, to help you better understand and organize your photo collections.

Retrieval



- Given a query image, find all images in the index set depicting the same landmark.
- Goal is to maximize Mean Average Precision (mAP) score - between 0 and 1 - based on number of correct and incorrect matches over a test set of query images.

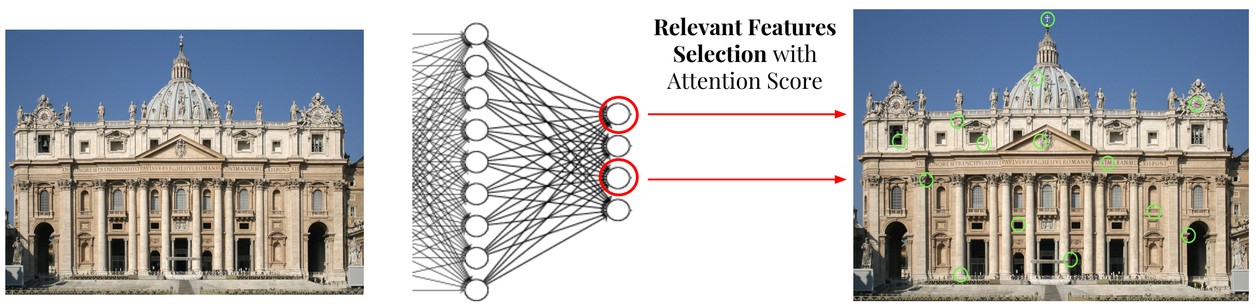
Recognition



- Given a query image, predict the landmark from ~15K classes.
- Additionally give a score corresponding to the confidence of the prediction.
- Goal is to maximize Global Average Precision score based on number of correct and incorrect predictions.

Previous Works: Deep Local Features Matching (DELF) [1]

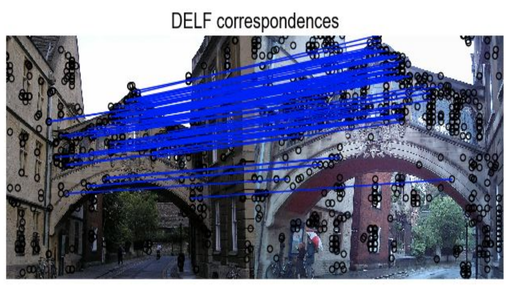
1) Deep Local Features Extraction



Dense feature extraction via Fully Convolutional Network

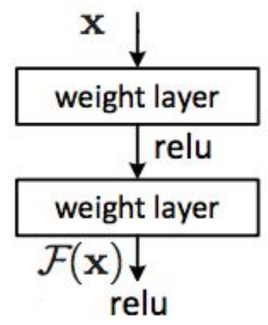
2) Local Features Geometric Matching

- Features are matched using **nearest-neighbor search** on a K-D Tree ('N' nearest-neighbors are a match)
- Matched features between two images are geometrically checked with **RANSAC algorithm**
- Images with **most geometrically agreeing features** (inliers) are considered to display the same landmark

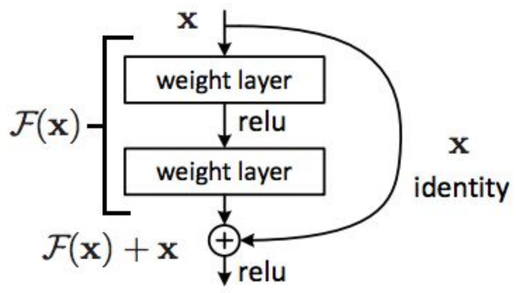


Previous Works: Residual Neural Network (ResNet) [2]

1) Residual Neural Networks



Plain Neural Network Architecture

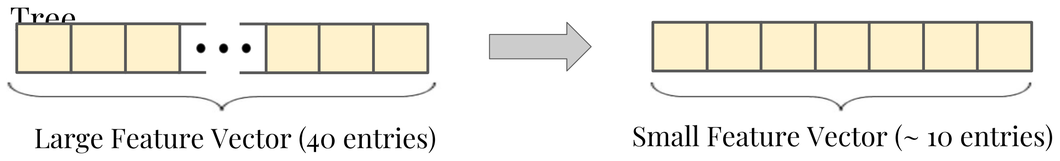


Residual Neural Network Architecture

- A residual neural network is similar to a plain neural network in all aspects except that the former has **"Shortcut" connections** to bypass certain weight layers.
- The residual networks has superior performance compared to a plain network because each layer "learns" only if there is something new to learn, hence the network doesn't tend to overfit.
- Such "shortcut" connections help in handling **vanishing gradient** problem and allow us to build deeper neural networks without the concern of overfitting or training error.

New Ideas and Results

- **Features Dimension Reduction** for fast Nearest Neighbor Search in K-D Tree

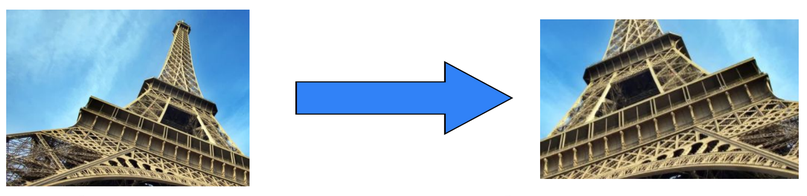


- **Increase number of nearest neighbors** (features matches) and nearest neighbor **distance threshold** to compensate for dimensionality loss
- **Inliers-to-Features Convex Hull Area Ratio** for robust geometric validation

Current mAP : **0.068**

Hyperparameter Tuning and Results

- Modified the final softmax layer to accommodate 14951 classes from the original 1000.
- Since there were few training data for certain classes, we randomly augmented images and copied them into the validation set to reduce overfitting.



- Different architectures were experimented:

Network	Training Accuracy (%)	Testing Accuracy (%)
ResNet18	26	25.3
Pre-Trained ResNet50	52	48
Fine-Tuned ResNet50	63	61.5

[1] Noh, H., Araujo, A., Sim, J., Weyand, T. & Han, B., '*Large-Scale Image Retrieval with Attentive Deep Local Features*'

[2] He, K., Zhang, X., Ren, S. & Sun, J., '*Deep Residual Learning for Image Recognition*'