Interview Challenge:
Computer Vision & Machine Learning for Image Retrieval Systems

Image Matching

The core functionality of Image Retrieval is based on the ability to identify matching images.

1) Assume I and J are images of the same painting in different resolutions, different aspect ratios, with and without compression artifacts, etc. What is your method of choice for confirming the two images show the same content? And why did you choose that particular approach?

2) Assume I and J are photos captured that same movie poster in different environments (Fig. 1) through a pin-hole camera. Name and describe an established method for image alignment (aka image registration) that would do a good job in minimizing the image error:

$$\min \epsilon = \sum_x \left| J(\mathbf{H}\tilde{\mathbf{x}}) - I(\mathbf{x}) \right|^2, \quad \mathbf{x} = (u \quad v)^\mathsf{T}, \ \tilde{\mathbf{x}} = (u \quad v \quad 1)^\mathsf{T}$$

$I(\mathbf{x})$ being the pixel intensities in image I at coordinate $\mathbf{x}$, and $J(\mathbf{H}\tilde{\mathbf{x}})$ the pixel intensities at the augmented coordinate $\mathbf{x}$ transformed by image transformation matrix $\mathbf{H}$. What type of transformation matrix $\mathbf{H}$ would be required to adequately model the mapping? Why is this error function formulation not ideal for image similarity in photos? What is a better alternative?

3) Name your favorite classic Image Feature Descriptor variant. Explain how it is computed. Why do you prefer it over the others?

4) How are visual feature-based 3D reconstruction technologies like "Structure-from-Motion" or robotics technologies like "Visual Odometry" related to Image Matching?

5) Assume J_i is a live camera image captured by a phone's camera (known camera intrinsics $\mathbf{K}$). And I is a reference image (known scale s), that is showing up in J_i's viewfinder (Fig. 2).

   Instead of exhaustively computing the precise image transformation between J_i and I for each frame, propose a light-weight method to estimate $\mathbf{H_i}$ by adjusting an initially estimated image transformation matrix $\mathbf{H}_0$ to track I using extra information provided for each frame index i :

   a) 3D motion $\mathbf{M}_i = [\mathbf{R}_i \ \ \mathbf{t}_i]$ obtained by the phone's non-visual motion sensors **or**

   b) 2D dense optical flow ($m_i : \mathbb{R}^2 \rightarrow \mathbb{R}^2$) obtained by the phone's hardware video codec **or**

   c) Propose your own method for target object detection & tracking on the phone

Fig. 1: A photo of a Blu-ray box cover and a public billboard showing the exact same movie poster content in a different environment

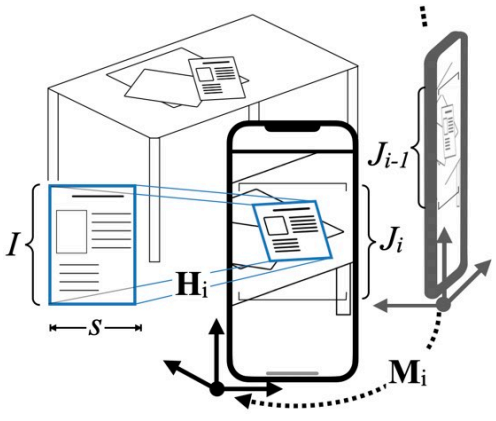

Fig. 2: Using a smart phone, J_i is continuously capturing the environment containing target I.

Image Indexing

For Large scale image retrieval (=reference image dataset sizes>100) it is impractical to use image matching one-by-one to find and retrieve a matching reference image. Instead, an image similarity search obtains a limited set (i.e. k<15) of potentially matching images in sublinear time.

Image Descriptors

Similarity search indexes all reference images using global Image Descriptor Vectors (sometimes called image hash or fingerprint) that abstractly capture the image's content in a number vector.

6) Describe the concept of the classic global Image Descriptor approach "Bag-of-Visual-Words". What are the visual words and how do they look like?

7) Describe the concept of a state-of-the-art global Image Descriptor approach using Convolutional Neural Networks (CNN). Describe the visual interpretation of that CNN's lower-level neuron weights.

8) How do the two approaches compare? What are the pros and cons of each over the other? Looking at challenging image similarity cases (Fig 3./4.), propose state-of-the-art techniques on how to overcome or mitigate the deficits of the CNN-based Image Descriptor approach.

Nearest Neighbor Search

A query image's Descriptor Vector is used to look up the k-nearest neighbor (KNN) vectors that represent similar dataset images by some distance metric.

9) Name and describe state-of-the-art data structures that are capable of storing large quantities of high dimensional vectors and support looking up nearest neighbors for a given query vector in sublinear time.
What are the particular trade-offs made by each data structure?

10) Image Descriptor Vectors of a dataset are typically highly correlated and greatly underutilize their high-dimensional parameter space.
What are the effects on K-Nearest-Neighbor data structures and how can this be mitigated?

Image Retrieval System

11) Define Content-base Image Retrieval (CBIR) in your own words and describe a full system design and it's component's / module's functionalities by combining Image Indexing and Image Matching. Name recent publications that cover modern CBIR.

12) How would such a system be designed to run as a cloud service for an app that captures query images through a phone's camera? How would the system's components be divided between cloud and app? What are the expected runtimes, bottlenecks and scalability issues?

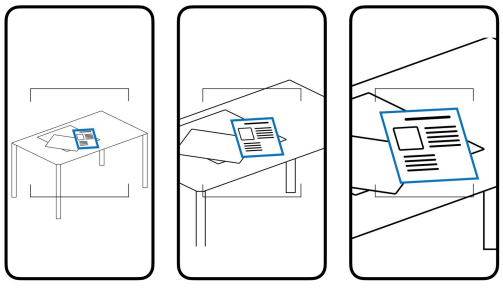

Fig. 3: Expected image acquisition cases of a target object using a hand-held smart phone

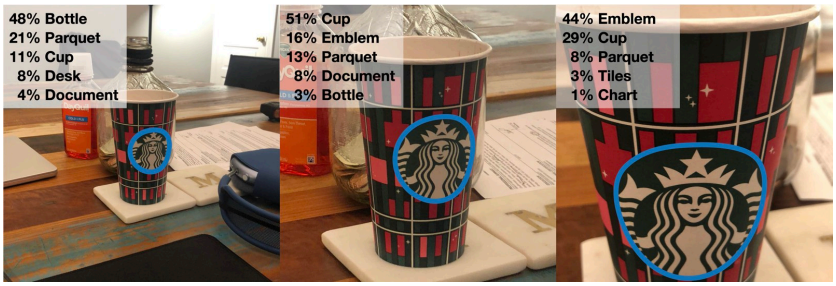

Fig. 4: Exemplary image classification results using a pre-trained neural network (i.e. ResNet50) of the same scenery containing a target image.