

Final Examination

---

**Exam Notes**

1. If doubt exists as to the interpretation of any question, the candidate is urged to submit the answer paper with a clear statement of any assumptions made.
2. This is an OPEN BOOK EXAM. Any paper notes or textbooks are permitted.
3. No calculator or computer or any sort is permitted.
4. This exam contains SIX (6) questions, however FIVE (5) questions constitute a complete exam paper. The first five questions as they appear in the answer book will be marked.
5. Each question is of equal value.
6. The clarity and organization of the answers are important.
7. The exam is three hours long.

**1. Basic Definitions**

For each of the following concepts, give a brief definition (2-4 sentences) and a description of where the concept is used or applied, in practice:

- (a) Lossy Compression
- (b) Anisotropic Diffusion
- (c) YUV Colour Space
- (d) Image Inpainting
- (e) Laplacian Pyramids
- (f) Image Segmentation
- (g) Robust Image Features, such as SIFT or SURF

**2. Image Segmentation**

Give short answers to each of the following:

- (a) Give a careful mathematical definition of image segmentation.
- (b) What sorts of assumptions are we making about the image, essentially its prior model, in doing segmentation?
- (c) Many methods have been proposed for image segmentation. Give the names of at least three methods, and summarize their respective assumptions, strengths, and weaknesses.
- (d) Where is image segmentation useful? Suggest three specific image processing applications that benefit from the use of image segmentation.
- (e) For each of the following circumstances, how does the circumstance add challenges to the segmentation process, and what steps might one need to do to have a successful segmentation?
  - Noisy images
  - Low contrast images taken at night
  - Colour images

### 3. The Frequency Domain

Give short answers to each of the following:

- (a) Give the equation / definition of the 2D discrete Fourier transform.
- (b) What are *advantages* in image processing of working in the frequency domain rather than the spatial domain?
- (c) Give examples of image processing problems which are effectively undertaken in the frequency domain, and explain why.
- (d) What are some *weaknesses* of doing image processing in the frequency domain?
- (e) Describe specifically how we can use frequency-domain methods to perform a spatial convolution.
- (f) What is the computational complexity of the 2D FFT? What is the essence behind its computational efficiency?
- (g) What is the computational complexity of the 2D wavelet transform? Exactly how is it that it is faster than the 2D FFT?

### 4. Image Matching

The problem of content-based image retrieval (CBIR) is of significant interest, since Google and other search engines attempt to produce relevant images based on search criteria. The challenge, of course, is that there are millions to billions or more images on the internet to search through.

Answer each of the following:

- a) What makes image searching fundamentally more difficult than searching text?

Really we have two related problems:

- Searching for *similar* images (e.g., all sunset pictures), or
- Searching for *matching* images (e.g., cropped, resized versions of the *same* original image).

- b) Give a succinct problem statement or mathematical definition of both problems.
- c) What sorts of parameters might a user specify in a query for each of these two problems?
- d) We would almost certainly want to extract different image features for these two problems. Suggest some possible features and briefly describe the rationale why a given feature is effective at finding similar or matching images.
- e) Suggest one or two strategies for making the similarity or matching search computationally efficient.

**5. Application**

Suppose that you have a quad-copter — a flying helicopter with four motors on which it is easy to mount a video camera. Because of wind and the movement / vibration of the quad-copter, the video sequence coming from the camera is somewhat jittery, causing the axis of the camera to move from frame to frame by up to 10 pixels.

We would like to be able to use the quad-copter to follow an object, such as a car.

**Your Task:**

Assume that there is a separate system for controlling the actual motion of the quad-copter, this is *not* your problem.

You need to design a system that tracks an object over time:

- A rectangle bounding the object of interest is provided to you in the first frame.
- In subsequent frames, track the object and return two things:
  - (a) Some description of the location and extent of the object.
  - (b) Some measure of the confidence that you have found the object. A confidence of zero means that the object can no longer be found and needs to be re-initialized.

Be sure to describe the system carefully, with a clear description for each system component and a clear outline of the entire system process.

You are providing a high level design: you do *not* need to produce a computer program, although you could choose to write pseudocode.

**6. Application**

You work for the geography department of a major Canadian city. The mayor of the city wishes to assess the photovoltaic (solar energy) potential of the city, and has asked you to come up with a strategy for measuring the total rooftop area of buildings and houses in the city.

Of course the most accurate approach would be have people take a survey, walking from building to building and measuring the roof area, however that would be far too expensive. Instead, you have good quality airplane / satellite imagery available, in colour at 10cm resolution, for the entire city. With this imagery you wish to try to measure roof area using image processing.

The imagery was acquired during the day time, near noon on a sunny day, in the summer. So lawns are green and shadows are minimized.

**Your Task:**

Describe your strategy, either in pseudo-code form or as a block diagram. You may also include sketches and other information to make your answer understandable to a programmer. State any assumptions you make.