Family Name ____________________

First Name ____________________

Student Number |__|__|__|__|__|__|__|__|

Venue ____________________

Seat Number ________

UNIVERSITY OF CANTERBURY
Te Whare Wānanga o Waitaha
CHRISTCHURCH NEW ZEALAND

No electronic/communication devices are permitted.

No exam materials may be removed from the exam room.

Computer Science and Software Engineering

EXAMINATION

Mid-year Examinations, 2019

COSC428-19S1 (C) Computer Vision

Examination Duration: 120 minutes

Exam Conditions:

Closed Book exam: Students may not bring in anything apart from writing instruments.

No calculators are permitted

Materials Permitted in the Exam Venue:

None

Materials to be Supplied to Students (if needed)**:**

- Extra sheets of write-on question paper (or answer book)

Instructions to Students:

- **Write your name and student ID above**
- This exam is worth a total of 100 marks
- Contribution to final grade: 40%
- Length: 10 questions
- Answer all questions.
- Check carefully the number of marks allocated to each question. This suggests the degree of detail required in each answer and therefore amount of time to spend on it.
- The amount of space provided also indicates the amount of detail expected.
- **Write strictly in the spaces allocated to each answer**. Do not write close to the margins, as the answer books will be scanned, and writing very close to the margin may not be picked up. If you require extra room, there is a blank page at the end of this booklet. You may also use additional sheets of paper; these must be fastened securely to your answer booklet. You should clearly indicate in the appropriate space that the answer is continued/provided elsewhere.

For Examiner Use Only

Question	Mark

Total ________

Questions Start on Page 3

1 [10 marks total)]

A good *edge* detector should have:

- Good Detection: filter responds to edge, not noise.
- Good Localization: detect edge near true edge.
- Single Response: one per edge.

(a) Describe how the Canny edge detection algorithm accomplishes the above attributes of a good edge detector. [6 marks]

(b) Explain how the choice of Gaussian kernel size affects the behavior of the Canny edge detector. [4 marks]

[[

(a)

Find gradient directions in an image:

The Canny edge detector uses a filter based on the first derivative of a **Gaussian**, because Canny is **susceptible to noise** present on raw unprocessed image data, so to begin with, the raw image is convolved with a Gaussian filter. The result is a slightly blurred version of the original which is not affected by a single noisy pixel to any significant degree.

An edge in an image may point in a variety of directions, so the Canny algorithm uses **four filters** to detect horizontal, vertical and diagonal edges in the blurred image. The edge detection operator (**Roberts, Prewitt, Sobel** for example) returns a value for the first derivative in the horizontal direction and the vertical direction. From this the edge gradient and direction can be determined. The edge direction angle is rounded to one of **four angles** representing vertical, horizontal and the two diagonals (0, 45, 90 and 135 degrees for example).

Steps of the Canny edge detection algorithm (using gradient directions found above):

1. **norm of the gradient** (i.e. along direction of line/curve)
2. **thresholding so as to respond to edges, not noise**
3. **thinning for good localisation and only one response per edge**. (for thinning, use non-maximum suppression = check if pixel is **local maximum** along gradient direction), **predict the next** edge point (assume the marked point is an edge point, then **construct the tangent to the edge curve** (which is normal to the gradient at that point) and use this to predict the next points)
4. **hysteresis to improve localisation**. (for hysteresis, check that maximum value of gradient value is sufficiently large. If drop-outs, use hysteresis **(using a high threshold to start edge curves and a low threshold to continue** them)),

(b)

The choice of Gaussian kernel size, σ, depends on desired behavior, where

- large σ detects large scale edges (and better noise suppression)
- small σ detects fine features

]]

2 [12 marks total]

A good local image *feature* to track should:

- satisfy brightness constancy
- have sufficient texture variation
- correspond to a "real" surface patch
- not deform too much over time

(Such good local image features are used for matching the same point in a stereo pair of images or in successive frames of video.)

Taking into account the above features, describe and compare the following two good local feature detection algorithms:

(a) Harris detector [6 marks]

(b) SIFT [6 marks]

[[

Harris:

- captures the structure of the **local neighbourhood** using a Auto-correlation matrix, where **2 strong eigenvalues** of this matrix indicate a good local feature (compared with 1 strong eigenvalue indicating a contour or no strong eigenvalues indicating a uniform region).
- Harris gives a **measure of the quality** of a feature – because the best **feature points can be thresholded** on the eigenvalues.

SIFT:

- **thresholded image gradients** are sampled over **16x16 array of locations** in scale space (at **8 different scales**/guassians)
- an array of **orientation histograms** is created at each location (e.g. 8 orientations x 4x4 histogram array = 128 dimensions)
- Because SIFT is based on a **vector of angles**, it is **computationally efficient**

Comparison:

SIFT and Harris are illumination and **rotation invariant** because they are based on operators on gradient, but **are not deformation invariant** (because deformations can change gradients).

SIFT is scale invariant because it is sampled at different scales, but **Harris is not scale invariant**.

They are translation invariant for x & y motion perpendicular to the camera - but not for z (scale changes).

]]

3 [14 marks total]

Describe how correctly matched feature points in two images enable finding:

(a) depth values in a stereo pair of images [5 marks]
(b) optical flow points in two successive frames of video using the Lukas Kanade algorithm [5 marks]
(c) Describe how depth can be calculated from optical flow. [4 marks]

(a)

One image is **rectified** (aligned) with respect to the other (using the "essential matrix").

Points lying on a **horizontal line** in one image are **matched with corresponding points** on the same line in the other image (.e.g. using **least squares** of pixel values over a region around each point).

The "*x*" distance between a matching pair of points is called the **disparity**. The **larger** the disparity, the **closer** is that point to the camera based on triangulation (but this is not linear).

(b)

Lucas-Kanade method **integrates gradients over a patch** to find features good enough to track using the **Harris** detector.

A **constant velocity** is assumed for all pixels within an image patch.

Optical flow is the measure of the movement that feature points undergo in successive frames.

(c)

Relative depth can be calculated from the **velocity of optical flow points** – which is larger when the depth is less. So absolute depth could be determined if the **velocity** is known.

Even for a camera moving forwards or backwards with no rotation - as depth decreases, the "**focus of expansion**" velocity increases (and vice-versa).

4 [16 marks total]

Briefly describe the following morphological operators and explain what effect they have on an image and why they have such an effect:

(a) Erosion [4 marks]
(b) Dilation [4 marks]
(c) Open [4 marks]
(d) Close [4 marks]

(a) Erosion: **Removes outside pixels** of a region/blob (and internal holes/regions) usually using a convolution kernel/mask in an *and* operation (or subtracts the *convolution of the kernel with the image*). Removes small details such as thin lines, noise points and widens gaps. Shrinks a region (to a skeleton with successive erosions).

(b) Dilation: **Adds pixels to the outside** of a region/blob usually using a convolution kernel/mask in an *or* operation (or adds the *convolution of the kernel with the image*). Enlarges a region/blob, thickens lines, fills small holes.

(c) Open: **Erode then dilate** image. (i.e. dilates an eroded image.)

Removes small details such as thin lines, spurs and noise. Smoothes jagged edges **without changing the size of the original object**.

(d) Close: **Dilate then erode image**. (i.e. erodes a dilated image.)

Closes/fills in small gaps/holes and preserves thin lines **without changing the size of the original object**.

5 [12 marks total]

In the context of computer vision based 3D reconstruction, briefly describe the following:

(a) Homography [4 marks]
(b) Essential matrix [4 marks]
(c) Bundle adjustment [4 marks]

[[

(a) Homography H: relates relative pose of 2 cameras viewing a planar scene. Estimate from feature correspondences using RANSAC.

(b) Essential matrix E: relates relative pose of 2 cameras viewing a 3D scene. Estimate from feature correspondences using RANSAC.

(c) Bundle adjustment (BA): initialise using RANSAC (for E), estimate a set of 3D points and camera poses which minimises reprojection error.

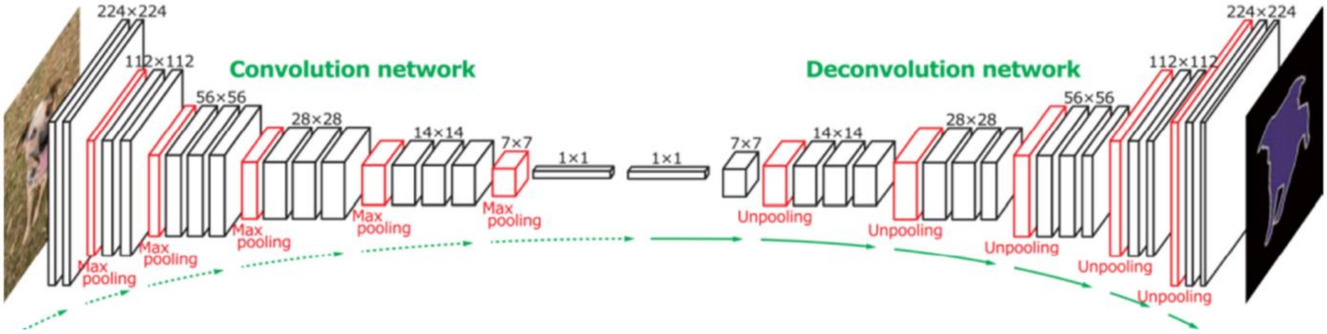
]]

6 [14 marks total]

Briefly describe the following four goals of deep learning applied to images:

(a) classification [3 marks]
(b) object detection [3 marks]
(c) dense segmentation [3 marks]
(d) instance segmentation [3 marks]

(e) State which of these four goals is achieved by the deep learning network below. [2 marks]



a) detect if an object is present or not in an image (I.e. not detecting where it is, but just detecting if such an object exists anywhere in an image.)

b) detect the location of an object in an image (returning region-of-interest/bounding-box coordinates)

c) label every pixel in an image as belonging to a class (such as grass pixels, sheep pixels)

d) label segmented pixels for each instance of a class (such as recognise which general sheep pixels in an image belong to which individual sheep for many overlapping sheep in a flock of sheep in an image)

e) instance segmentation (or dense segmentation)

7 [6 marks total]

Deep learning (CNN) is a game changer in computer vision – and encompasses different learning approaches including *unsupervised* learning, *supervised* learning and *reinforcement* learning. Give an example for each of these *three* learning approaches. [2 marks each]

unsupervised learning: graph cut

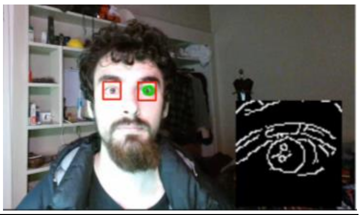
supervised learning: supply annotated images

reinforcement learning: interact with environment to evaluate performance

8 [16 marks total]

You are to briefly describe **only four of the following** class projects [for 4 marks each] by just listing (one per line) at least four algorithmic steps, **naming the algorithms** used in the order they were used.

Do not select your own or similar project (e.g. face recognition projects - do not select other face recognition projects, etc).



(a) “Fatigue Detection” by locating eyes to detect blink rate.

Haar Cascade

Gaussian Blur

Canny Edge Detection

Hough Circle Transform

(b) “Billiards Top-Down Perspective Transform”

HSV Colour Space Transform

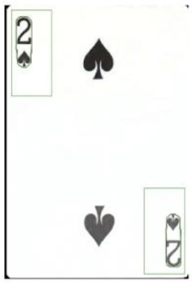
Threshold on Colour value

Canny Edge Detection

Perspective Transform

(Image Cropping)

(c) “Card Recognition”



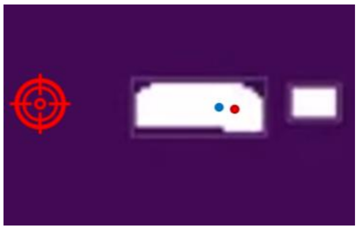
Dataset Generation

- Gaussian Filter
- Canny Edge
- Contour Detection
- Image Warping
- Convex Hull Detection

Object Detection

- You Only Look Once (YOLOv3)

(d) “Predictive Animal Tracking for Predator Identification”



Background Subtraction

Thresholding

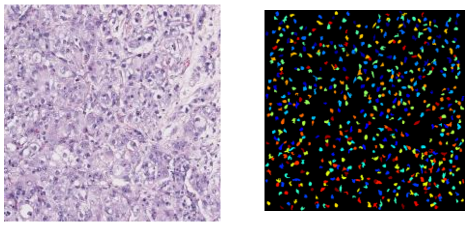
Erosion and Dilation (Opening)

Moments of Image

Ordinary Kalman Filtering

Moving Average filtering

(e) “Cell Segmentation from Breast Cancer Whole Slide Images”



HSV conversion (and Normalise channels to use full value range)

Threshold

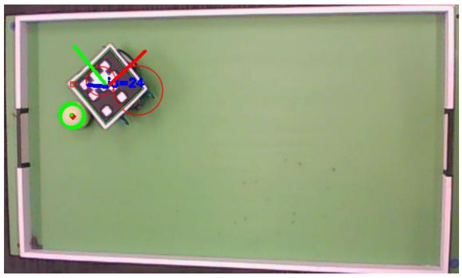
Gaussian Blur

Erosion and Dilation

(CellProfiler application)

(Blob detection)

(f) “Robot Soccer” to detect ball and robot.



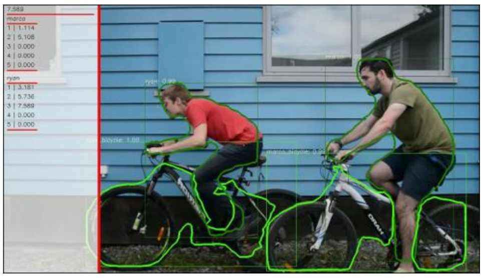
Ball detection:

- Hough Circles
- Differencing
- Blob Detection
- Colour Segmentation + Kalman
- HSV Segmentation

Robot Detection:

- Fiducial Markers

(g) "Cyclist Detection and Identification"

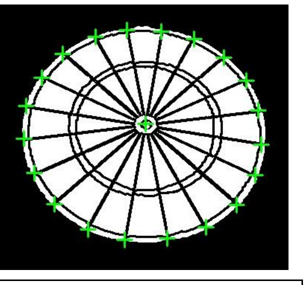


Mask R-CNN Neural Network with COCO Dataset

K-means Pixel Clustering

RGB Thresholding

(h) "Automatic Dartboard Scoring"



Difference algorithm

Hough transform algorithm

Floodfill algorithm

Dilation and Erosion algorithm

Opening and closing algorithm

Canny edge algorithm

Moments algorithm (finding the centre of threshold)

Contour generation algorithm

Random Gaussian noise algorithm

… extra space …

If you use this page, please refer to it from the original question.

End of Examination