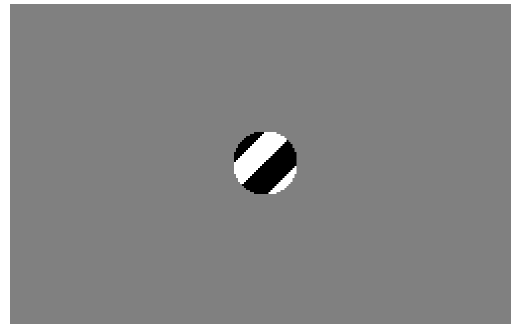
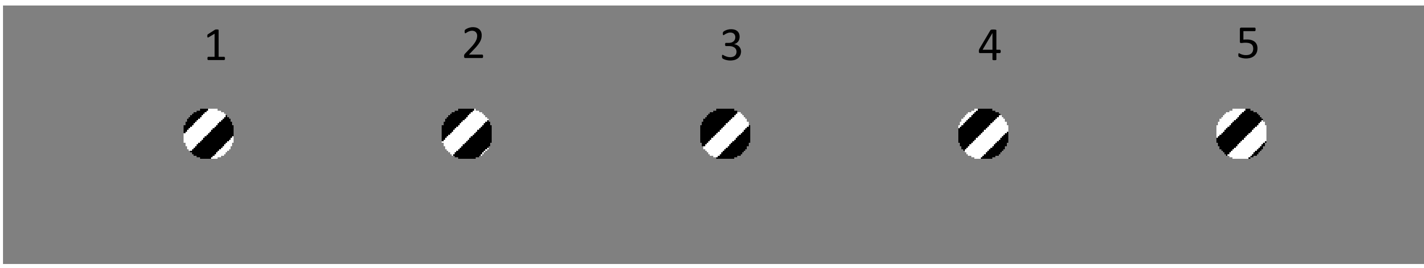
CS4243 Quiz 4

Solutions given in red boxes. Dashed red boxes are auto-graded by LumiNUS. Solid red boxes graded manually by TA.

Multiple Choice (2)





Observe the above GIF of moving grates. Sample frames are provided in numerical sequence below. How many unique motions could generate the above effect?

(a) 1

(b) 2

(c) 3

(d) 4 or more (aperture problem, can be determined only uniquely to a line)

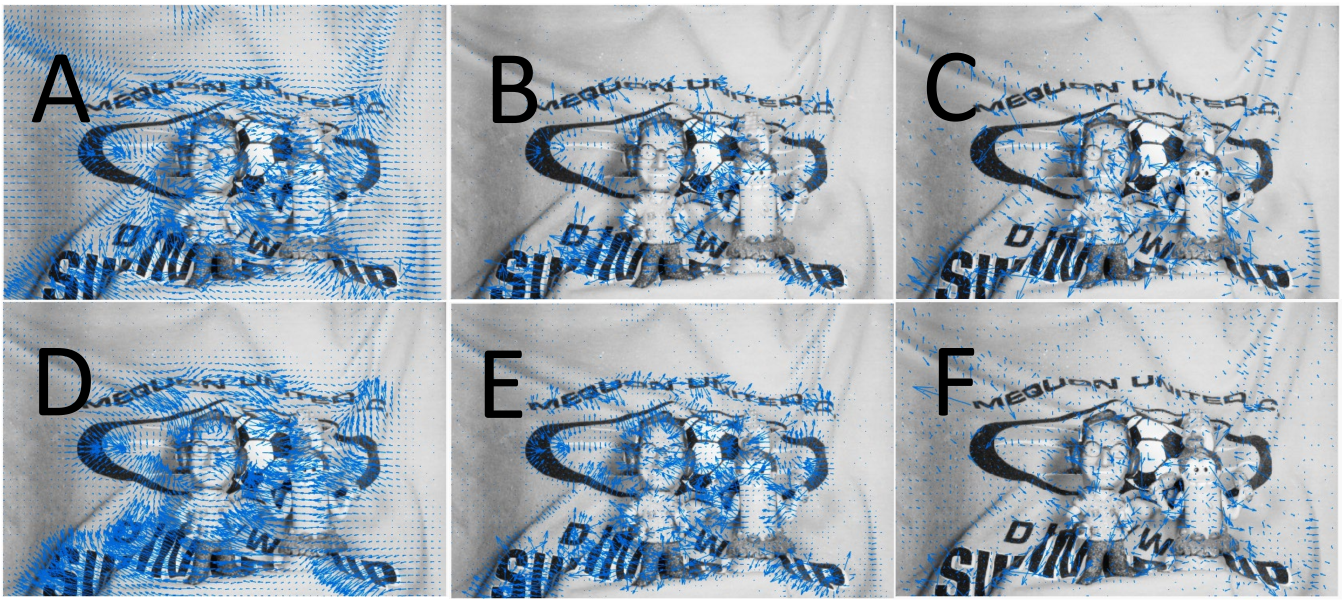
$$H = \sum_{\mathbf{x}} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right]^{\mathrm{T}} \left[\nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right]$$

Consider the above definition of the (approximated) Hessian matrix used in the Lukas-Kanade algorithm. If we consider an affine warp model and a template of N pixels, what is the computational complexity of estimating the Hessian given in big-O notation?

O(6^2 * N) = O (36N)

Optical Flow (3)

HS high lambda, 1000 iters | HS low lambda, 50 iters | LK High thresh



HS low lambda, 1000 iters | HS high lambda, 50 iters | LK Low thresh

- Lukas-Kanade has sparse optical flow, HS has dense optical flow with smooth flow fields
  - Sparse-looking: B, C, F. Rule out B since flow vectors are still regular; so just C and F remaining. F is noisier and more flow vectors, so has the lower threshold for R;
- Of remaining A, B, D, E for Horn-Shunck; A and D have more iterations – cannot recover the fine movements of the low-gradient areas such as the fabric background without running many iterations
  - Smaller lambda, smoother flow-fields so A is large lambda, B locally smoother than E.

Deep Learning (5)

1. You have a binary classification problem with input $\boldsymbol{x} \in R^{100 \times 1}$. You consider designing a multi-layer perceptron (MLP) with two hidden layers and one output layer. Each hidden layer perceptron unit has no bias terms; the output layer consists of a single perceptron also with no bias. You consider:

- MLP A has 50 units in the first hidden layer and 10 units in the second hidden layer.

- MLP B has 10 units in the first hidden layer and 50 units in the second hidden layer

How many parameters does each MLP have?
MLP A: (numerical entry)
100 x 50 + 50 x 10 + 10 = 5 000 + 500 + 10 = 5510
MLP B: (numerical entry)
100 x 10 + 10 x 50 + 50 = 1000 + 500 + 50 = 1550

(2) Which one of the following use cases exemplifies the above-described problem? (MC)
we are looking for an input of 100 dimensions and a binary classification output

- An email filter which flags spam based on 50 commonly used heuristics. *(Input of 50 dimensions; output is binary classification)*
- A numerical digit classifier which takes as input 10x10 pixel images. *(Input of 100 dimensions; output is 10 classes so not binary)*
- A weather app which predicts the amount of rainfall tomorrow based on the past 100 days. *(Input of 100 dimensions; output is regression)*
- An automated stock trading system which buys or sells a fixed unit of stocks based on the share prices of the 100 largest companies listed on Singapore exchange. *(Input of 100 dimensions; output is binary classification)*

(3) Which of the following statements is incorrect?
- Deep learning models are trained over large-scale datasets
- Deep learning has been fueled by the improvements of hardware like GPUs
- Deep learning is a rule-based AI solution (statistically learned, no rules set, see L12, slide 11)
- Deep learning excels at narrow AI problems such as recognizing street signs and translating Chinese to English text

(4) You wish to estimate the weights for a simple perceptron model defined as follows.

$$\tilde{y} = \begin{cases} 1, & if\ x_1 w_1 + x_2 w_2 + b > 0 \\ 0, & else \end{cases}$$

Which set of weights is valid for all 4 of the data samples presented in the table?

x1	x2	y
1	1	1
1	0	1
0	1	1
0	0	0

A. w1=1, w2=1, b=0.01
B. w1=0.1, w2=0.1, b=-0.5
C. w1=1, w2=1, b=-0.5
D. w1=0.1, w2=1, b=-0.5

System Design: Speeding Vehicles (11)

The Ministry of Transport asks you to design a computer vision solution for detecting speeding vehicles. You are provided footage from a camera which is mounted on an overpass and captures images similar to the figure below. You are asked to find all the cars in the scene and give an estimate of their driving speed. Those with speeds too high should be flagged.

(i) Outline a method to find the individual vehicles using any of the techniques we have studied in lecture to date, clearly stating any assumptions you make.

- Motion segmentation via flow patterns; assumes cars do not drive too “close” in scene (this doesn’t work well in as it fades into the background)
- colour-based segmentation; assumes that cars do not have a similar colour as the street
- 0.5 point for reasonable method; 0.5 point for assumption

(1)

(ii) Assuming that you are able to find the individual vehicles in the scene, outline a method to estimate each vehicle’s velocity using optical flow. Clearly state any assumptions you make. Comment on the accuracy of your approach and what can be done to improve the accuracy.

Method	Assumption	Accuracy
Compute optical flow in region around each vehicle, mark those whose average optical flow magnitude exceeds some threshold	Optical flow magnitude is proportional to vehicle’s velocity	accuracy is poor because optical flow estimation is often not accurate and can also be affected by camera movement; can improve by make threshold proportional to location in the scene since optical flow is given in pixel units / frame

0.5 point for reasonable method, 0.5 point for assumption, 1 point for accuracy & improvements (2)

(iii) You find that your system is not accurate in estimating driving speeds because the camera shakes when there are strong winds or when large vehicles drive by on the overpass. Outline how you can estimate the camera movement and how to compensate for camera movements in your estimated solution (ii), clearly stating your assumptions.

- (method 1) estimate global flow field (not include vehicles); subtract optical flow from local optical flow of the vehicles; subtract this flow from the vehicle flow; this method assumes that camera motions are much smaller than that of the vehicles
- (method 2) Estimate motion from frame to frame via feature matching on areas which do not belong to the vehicles, then estimate homography via DLT and warp one frame to next – assumes that motions are projective and qualifies for DLT conditions (approximately planar scene / rotational camera movements). Post-warping, estimate velocity of vehicle again, or directly apply projection to the locations before computing the velocity

0.5 point for reasonable method; 0.5 point for assumption; 1 point for compensation

(2)

System Design: Speeding Vehicles (cont'd)

(iv) Upon closer analysis, you observation that the camera motion described in (iii) is affine and can be parameterized by p_1 … p_n. Based on this observation,

- Define the equations for the flow components u and v

$$u(x, y) = p_1 x + p_2 y + p_3$$
$$v(x, y) = p_4 x + p_5 y + p_6$$

- Define the brightness constancy equation incorporating this motion model, using I_x, I_y and I_t to indicate the image gradients in space and time

$$I_x(p_1 x + p_2 y + p_3) + I_y(p_4 x + p_5 y + p_6) + I_t = 0$$

- If your images are of resolution w x h, what is the minimum number of pixel for which you would need to estimate motion? 2, 3, 6, m x h, not enough information
- Each pixel provides 1 equation; 6 unknowns, so need 6 equations

(3) 1 point per sub-question

(v) You decide to find speeding vehicles with the mean-shift tracker from lecture 11. To initialize your tracker, you apply a car detector which returns tight bounding boxes around the detected vehicle. For traffic approaching the camera, you detect cars on the top half of the frame and track them as they drive towards the camera. For traffic moving away from the camera, you apply the detector on the bottom part of the frame and track the cars as they drive away.

You find that your tracker is not very accurate and either drifts off of the vehicle as you track it throughout the scene or jumps onto other vehicles in adjacent lanes. Why does this happen and how can you correct for the problem?

(2)

In this traffic set-up, there are extreme changes in scale as the car drives either towards or away from the camera and the basic mean shift tracker as defined in the lecture notes does not account for changes in scale.
Possibilities to correct include:
(1) Adjust the bounding box manually with some fixed percentage as it moves away from (decrease size) or towards (increase size) the camera
(2) Scale-sensitive mean shift tracker (add scale as one of the unknown parameters)
(3) Warp the scene so that the car size remains constant before applying the tracking algorithm

1 point for scale issues, 1 point for reasonable solution

System Design: Panorama-Gone-Wrong (6)

(i) The above two images are examples of panoramas gone wrong on smart-phones. Describe why this might happen.

The stitching together of the panorama correctly matches and stitches the stationary background but not the moving foreground object.
0.5 points for mentioning moving foreground object;
0.5 points for mis-match in stitching.

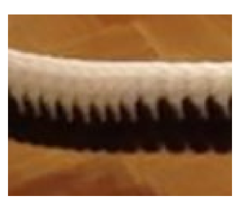
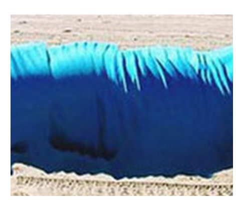
(1)

(ii) Suppose you want to achieve the same effect shown in the images above on purpose. You are given a video or series of images as input and you should return a single image as output. Design a computer vision pipeline to accomplish this. For each stage of the pipeline, specify the inputs, outputs, method used, as well as any assumptions made. (3)

Stage & Aim	Input	Output	Algorithm	Assumption
1. Separate foreground / background	series of images / extracted frames from video	Masks isolating object from background	Optical flow + motion segmentation e.g via clustering	Foregound object moves more and differently than the background;
2. Compose background	series of images of background with foreground object removed via masks in stage 1	Single composite image of background	DLT / homography + stitching	Constraints for using homography: scene is (approximately) planar / captured only under camera rotation
3. Compose foreground object	series of images of foreground with background removed based on masks found in stage 1	Single composite image of object	Cropping of object from each frame + overlay	Foreground object does not move erratically; object's dominant motion is a single translation across the scene at a (approximately) constant velocity

Per stage, 0.5 point for aim / input / output; 0.5 point for alg + assumptions

(iii) From the provided crops of the panoramas, we can see that the composed objects have an accordion-like effect in the middle of its body. Outline a method to ensure that your composite foreground object achieves the same effect, stating any assumptions made.

Track (specific) features within the foreground object from scene to scene via KLT tracking. To ensure that the effect is semi-regular, we should then crop the same small region in the middle part of the body from each frame to compose the foreground object. This assumes that the body has features that meets criteria for tracking, i.e. sufficient corners present
(1 point for algorithm & regularity, e.g. feature tracking & stitching, 1 point for assumptions, e.g. sufficient corners on body).

(2)