Midsem: CS663 Fall2024

Instructions: There are 2 hours for this exam. Answer all 7 questions, each of which carries 10 points. This exam is worth 20% of the final grade. Some formulae are listed in the beginning. **Think carefully and write *concise* answers on the question paper itself and ONLY in the designated places. You may use additional sheets for rough work, but you need to attach them behind this question paper.** You can quote theorems/results proved in class without re-deriving them.

Name: ____________________

Roll number: ____________________

Signature: ____________________

Some formulae:

1. Gaussian pdf in 1D centered at $\bar{x}$ and having standard deviation σ: $p(x) = \frac{1}{\sqrt{2\pi}\sigma}e^{-(x-\bar{x})^2/(2\sigma^2)}$. Multivariate Gaussian: $p(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{d/2}\sqrt{|\boldsymbol{\Sigma}|}}\exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^t\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$ for $\boldsymbol{x} \in \mathbb{R}^d$, mean vector $\boldsymbol{\mu} \in \mathbb{R}^d$ and covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{d\times d}$.

2. 1D Fourier transform and inverse Fourier transform (x and μ are spatial and frequency coordinates respectively):
$F(\mu) = \int_{-\infty}^{+\infty} f(x)e^{-j2\pi\mu x}dx, f(x) = \int_{-\infty}^{+\infty} F(\mu)e^{j2\pi\mu x}d\mu$

3. 2D Fourier transform and inverse Fourier transform ((x,y) and (μ,ν) are spatial and frequency coordinates respectively):
$F(\mu,\nu) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y)e^{-j2\pi(\mu x+\nu y)}dxdy, f(x,y) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} F(\mu,\nu)e^{j2\pi(\mu x+\nu y)}d\mu d\nu$

4. Convolution theorem: $\mathcal{F}[(f*g)(x)](\mu) = F(\mu)G(\mu); \mathcal{F}(f(x)g(x))(\mu) = (F*G)(\mu)$ where $\mathcal{F}$ is the Fourier operator.

5. Fourier transform of $g(x-a)$ is $e^{-j2\pi ua}G(\mu)$. Fourier transform of $\frac{df^n(x)}{dx^n} = (j2\pi\mu)^n F(\mu)$ ($n > 0$ is an integer). Also, we have $\int_{-\infty}^{+\infty} f(t)g^*(t)dt = \int_{-\infty}^{+\infty} F(\mu)G^*(\mu)d\mu$ where $*$ here is the complex conjugate.

Questions

1. Consider two 1D images $f(x)$ and $g(x)$ such that $g(x) = f(x-x_0)$. Both images are defined on a continuous domain and are continuous-valued. [3+3+4=10 points]

   (a) If the respective 1D Fourier transforms of the two images are $F(\mu)$ and $G(\mu)$, then write down an expression for $H(\mu) := \dfrac{F(\mu)G^*(\mu)}{|F(\mu)||G(\mu)|}$ in terms of x_0. Here $|F(\mu)|$ is the magnitude of the Fourier coefficient $F(\mu)$. Recall: if $a+jb$ is a complex number, then $a+jb = \sqrt{a^2+b^2}(\cos\theta + j\sin\theta)$ where $\tan\theta = b/a$.
   Answer: $H(\mu) = e^{j2\pi\mu x_0}$.

   (b) Write down an expression for the 1D inverse Fourier transform of $H(\mu)$.
   Answer: $h(x) = \delta(x-x_0)$.

(c) Write down the practical application of the result you see in the previous two sub-problems (describe in NOT more than 2 sentences)
Answer: The practical application is in image alignment to obtain translation. Given two images, you would first compute $h(x)$ via $H(\mu)$ as defined above, and then the peak of $h(x)$ will occur at the true shift value between the two images. **Marking scheme:** You need to mention that you look for a peak in $h(x)$, otherwise two points are deducted. Also note that this method can take care of only translation.

2. Consider a noisy image J such that $J(x,y) = I(x,y) + Z(x,y)$ where $Z(x,y) \sim \mathcal{N}(0,\sigma^2)$ is independent identically distributed Gaussian noise and I is a clean image. Assume both J and I contain N pixels. [3+3+4=10 points]

   (a) For any (x,y) in the image domain, $p(J(x,y)|I(x,y)) = \underline{\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(J(x,y)-I(x,y))^2}{2\sigma^2}}}$.

   (b) A student has an idea for a new noise removal technique. The idea is to generate a fresh set of N noise values from $\mathcal{N}(0,\sigma^2)$ and deduct these values from J to yield the supposedly denoised image K. Write down an expression for the PDF of the noise values in K (you need not simplify to the last step). $\underline{(g \otimes g)(\alpha)}$ where $g(\alpha) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{\alpha^2}{2\sigma^2}}$ and where $\otimes$ stands for the correlation operator. Since $g(\alpha)$ is symmetric about 0, the correlation here reduces to convolution, so that $(g * g)(\alpha)$ is also a correct answer with $*$ standing for convolution.

   (c) Is the student's idea an effective denoising technique? Explain why (not) in at most 2 sentences.
   Answer: The student's idea is not useful because doing so actually increases the noise variance from σ^2 to $2\sigma^2$.

3. (a) Given a noisy image J, consider the following two different operations: (1) You convolve J with a Gaussian and then compute the Laplacian of the result to yield image K_1. (2) You compute the Laplacian of J and then convolve the result with a Gaussian to produce image K_2. Assuming all images are of infinite extent (or ignoring border issues), will K_1 and K_2 be different? Explain in at the most two sentences. [5 points]
   Answer: Due to commutativity of convolution, we have $K_1 = K_2$.

   (b) Consider an image processing task which requires applying the Laplacian of Gaussian filtering transformation to an image $f(x,y)$, i.e. it requires computing $\nabla^2(G_\sigma * f)(x,y)$ where G_σ stands for a Gaussian signal with mean zero and standard deviation σ. In practice, what is the reason why masks of $\nabla^2 G_\sigma$ for different values of σ are pre-computed instead of computing $\nabla^2(G_\sigma * f)$ on the fly? Answer in not more than 2-3 sentences. [5 points]
   Answer: Masks for $\nabla^2 G_\sigma$ are pre-computed for the sake of computationally efficiency instead of computing the full set of two convolutions on the fly. Also doing so is correct due to the associative nature of convolution. **Marking scheme:** 2.5 points to be deducted if the associative nature of convolution is not mentioned.

4. Consider a signal processing system which takes an input signal $f(t)$ and outputs its Fourier transform $F(\mu)$. Answer the following questions: [3+3+2+2=10 points]

   (a) Is this a linear system? Give reasons in not more than 2-3 sentences.
   Answer: This is a linear system because we know that the Fourier transform is a linear transform.

   (b) Is this a shift-invariant system? Give reasons in not more than 2-3 sentences.
   Answer: This is not shift invariant because $\mathcal{F}(f(x-x_0))(\mu) \neq F(\mu - x_0)$ where $F(\mu) = \mathcal{F}(f)(\mu)$.

   (c) Let $G(\mu)$ be the Fourier transform of a signal $g(t)$. Suppose you have computed $G(\mu)$ during some calculations, but do not know $g(t)$. How will you compute $\int_{-\infty}^{+\infty} g(t)dt$ *directly* from $G(\mu)$

without obtaining a complete inverse Fourier transform?
Answer: We see that $G(0) = \int_{-\infty}^{+\infty} g(t)e^{j2\pi\mu(0)}dt = \int_{-\infty}^{+\infty} g(t)dt$.

(d) Continuing the earlier part, how will you compute $\int_{-\infty}^{+\infty} |g(t)|^2 dt$ *directly* from $G(\mu)$ *without* obtaining a complete inverse Fourier transform?
Answer: By Parseval's theorem, we know that $\int_{-\infty}^{+\infty} |g(t)|^2 dt = \int_{-\infty}^{+\infty} |G(\mu)|^2 d\mu$.

5. (a) State one advantage and one disadvantage of local histogram equalization over global histogram equalization. (no more than one sentence each) [2+2 = 4 points]
Answer: LHE produces better contrast than GHE. But LHE is more computationally expensive than GHE. Also LHE can amplify noise more than GHE.

(b) A computer monitor displays images by outputting voltages whose values are of the form $v = cr^\gamma$ where r is the input intensity, and $c > 0, \gamma > 0$ are properties of the monitor. Thus the monitor will display a brightness value cr^γ instead of the intended r. To correct for this, the input intensities need to be pre-processed before inputting to the display system of the computer. The mathematical expression for this pre-processing step is given by $s(r) = \underline{(r/c)^{1/\gamma}}$ for input intensity r. Write a one-line proof to show that the output voltage after this pre-processing will indeed be r: $\underline{c[(r/c)^{1/\gamma}]^\gamma = r}$. [3+3=6 points]

6. (a) Consider you want to do mean-shift based clustering on a set of 1D values $\{x_i\}_{i=1}^n$, for which you build a kernel density estimator $\hat{p}_N(x;\sigma) = \frac{1}{N}\sum_{i=1}^N \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-(x-x_i)^2/(2\sigma^2)\right)$ with bandwidth parameter σ. Given an initial value x, write the expression for its mean shift vector. [4 points]
Answer: The mean shift vector is given by $\dfrac{\sum_{i=1}^N x_i \exp\left(-(x-x_i)^2/(2\sigma^2)\right)}{\sum_{i=1}^N \exp\left(-(x-x_i)^2/(2\sigma^2)\right)} - x$.

(b) Write the index number of all true statements from those below. For example, if only the first two statements are true, then your answer will be i, ii. You will get marks only if all true and no false statement is/are chosen. $\underline{i, ii, iv}$ [3 points]

i. The evolving coordinates during the mean shift updates starting from some value x converge to a local mode of the probability density. This is a key property of mean shift taught in class.

ii. The probability density at the evolving coordinates during mean shift updates starting from some value x monotonically increases until it reaches the mode. This is a key property of mean shift taught in class.

iii. The convergence guarantees of mean shift are true for any kernel function including $K(x) = x$. The convergence is only for symmetric kernels which decay towards $\pm\infty$ as stated in the slides.

iv. Imagine you ran mean shift updates till convergence on all points in your dataset $\{x_i\}_{i=1}^n$ reaching coordinates $\{x'_i\}_{i=1}^n$ respectively. If you decided to run mean shift again, starting from $\{x'_i\}_{i=1}^n$, the point coordinates will not change compared to $\{x'_i\}_{i=1}^n$. When you get convergence in mean shift, you have reached the local mode of the density where the density gradient is 0. Hence running meanshift on the converged points will not cause any change in their coordinates. This statement is therefore false.

(c) A bilateral filtering operation on image $I(x,y)$ to produce $J(x,y)$ is given by:
$$J(x,y) = \frac{\sum_{(x',y')\in N(x,y)} I(x',y') \exp\left(-\frac{(x-x')^2+(y-y')^2}{2\sigma_s^2} - \frac{(I(x,y)-I(x',y'))^2}{2\sigma_I^2}\right)}{\sum_{(x',y')\in N(x,y)} \exp\left(-\frac{(x-x')^2+(y-y')^2}{2\sigma_s^2} - \frac{(I(x,y)-I(x',y'))^2}{2\sigma_I^2}\right)}.$$
Here $N(x,y)$ is a neighborhood around (x,y), and σ_s, σ_I are the spatial and intensity smoothing parameters. If σ_I is chosen to be very large (infinity), then which of the following filters will the bilateral filter be (approximately) equivalent to? For example, if only the first two options are correct, then your

answer will be i, ii. Write down the index numbers of all correct and no wrong options to get marks for this. *i* [3 points]

i. Weighted mean filter with Gaussian weights having mean 0 and standard deviation σ_s
ii. Weighted mean filter with Gaussian weights having mean 0 and standard deviation σ_I
iii. Median filter
iv. Sobel filter
v. Mean filter
vi. Identity filter (i.e. no filtering operation occurs)

7. **Image de-tearing:** Imagine you are showing your family the best player's certificate you won in an inter-IIT cricket tournament! But your two-year old nephew innocently tears it into two pieces. You decide to use your image processing knowledge to produce a seamless image of the certificate. For this, you scan the two pieces via a flat-bed scanner to yield two different rectangular images (*e.g*, see Fig. 1). You then want to align these two images, that is "digitally place the two pieces side by side". Following this, you can then do some kind of interpolation/filtering across the torn borders to get a smooth image. For the alignment step, you extract the edges at the torn borders in both images and accurately mark out n pairs of physically corresponding points $\{(x_{1i}, y_{1i})\}_{i=1}^n$ and $\{(x_{2i}, y_{2i})\}_{i=1}^n$ on the two borders using a tool in MATLAB. Answer the following questions. **For simplicity, assume that the two pieces if joined together would produce a complete image of the certificate and that there are no missing parts**: [5+5=10 points]

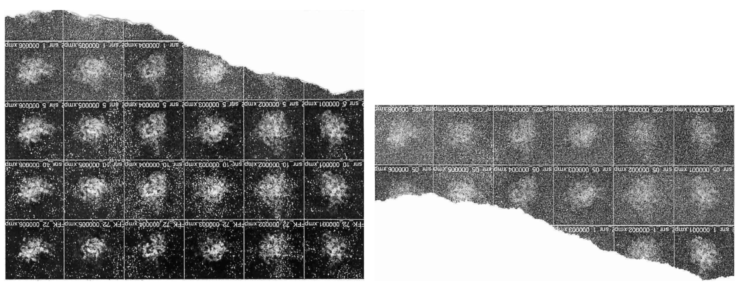

Figure 1: Scanned images of torn pieces of a document

(a) What motion model accurately expresses the relationship between (x_{1i}, y_{1i}) and (x_{2i}, y_{2i})? Ignore errors in marking coordinates. Express the relationship using a mathematical equation (use homogeneous coordinates if required) with all terms clearly defined.
Answer: The motion model involves rotation and translation. Here scaling and shearing are not required and 2.5 marks will be deducted for needlessly mentioning them. The equation is:
$$\begin{pmatrix} x_{2i} \\ y_{2i} \\ 1 \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta & t_x \\ -\sin\theta & \cos\theta & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{1i} \\ y_{1i} \\ 1 \end{pmatrix}.$$

(b) Instead of using manually marked out control points, suppose you wanted to use an intensity-based method to perform the alignment of both pieces. Write down the cost function you will minimize w.r.t. the motion parameters. Define the meaning of all terms in the cost function.
Answer: Here the main aim is to minimize the intensity differences near the border where the tearing occurred. Consider corresponding points (x_{11}, y_{11}) and $(\tilde{x}_{21}, \tilde{y}_{21})$ lying on the border of the two images respectively. Here $(\tilde{x}_{2i}, \tilde{y}_{2i}, 1)^t = M_{\theta,t_y,t_x}(x_{2i}, y_{1i}, 1)^t$ is a motion-transformed version of the point $(x_{2i}, y_{2i}, 1)^t$.
We now consider the local perpendiculars $(l_{11,x}, l_{11,y})$ and $(l_{21,x}, l_{21,y})$ at each border point on the two images, and compute the intensity difference $(I(x_{11}+l_{11,x}, y_{11}+l_{11,y}) - I(\tilde{x}_{21}+l_{21,x}, \tilde{y}_{21}+l_{21,y}))^2$

between points on either side of the border. The corresponding cost function is: $\sum_{i=1}^{n}(I(x_{1i} + l_{1i,x}, y_{1i} + l_{1i,y}) - I(\tilde{x}_{2i} + l_{2i,x}, \tilde{y}_{2i} + l_{2i,y}))^2$.