CARDIFF UNIVERSITY
EXAMINATION PAPER

Academic Year:	2012-2013
Examination Period:	Autumn
Examination Paper Number:	CMT107
Examination Paper Title:	Visual Computing
Duration:	2 hours

Do not turn this page over until instructed to do so by the Senior Invigilator.

Structure of Examination Paper:

There are **4** pages.

There are **4** questions in total.

There are no appendices.

The maximum mark for the examination paper is **72** and the mark obtainable for a question or part of a question is shown in brackets alongside the question.

Students to be provided with:

The following items of stationery are to be provided:

ONE answer book.

Instructions to Students:

Answer **3** questions.

The use of a translation dictionary between English or Welsh and another language, provided that it bears an appropriate school stamp, is permitted in this examination.

1. Vectors, Transformations and Modelling

(a) Given two 3D orthogonal vectors **a** and **b** which both have a length of 3 units, what is the *value* of $\mathbf{a} \cdot \mathbf{b}$, and what is the *length* of $\mathbf{a} \times \mathbf{b}$? Explain your reasoning. [4]

(b) A *homogenous coordinate* representation of a point is **p** = (6, 2, 4). What is its *Cartesian coordinate* representation? [3]

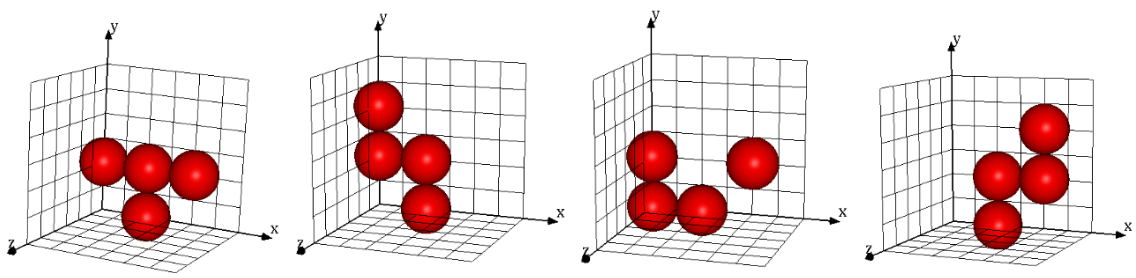
(c) List *three* different reference frames used in graphics. [3]

(d) A polygon must be *simple* if you want to draw it correctly using OpenGL. What does it mean for a polygon to be *simple*? What *other two* properties (give names and definitions) are required for a polygon to be drawn correctly using OpenGL. [6]

(e) *Which* of the images displayed below is created by the following code segment? Justify your answer. (Note that all the sphere centres have z-coordinate 0.) [8]

Total: [24]

```
glTranslatef(2.0, 0.0, 0.0);   //Line 1
glutSolidSphere(1.0, 32, 32); //Line 2
glPushMatrix();               //Line 3
glTranslatef(2.0, 2.0, 0.0);   //Line 4
glutSolidSphere(1.0, 32, 32); //Line 5
glPopMatrix();                //Line 6
glTranslatef(-2.0, 0.0, 0.0); //Line 7
glutSolidSphere(1.0, 32, 32); //Line 8
glTranslatef(0.0, 2.0, 0.0);   //Line 9
glutSolidSphere(1.0, 32, 32); //Line 10
```



(A) (B) (C) (D)

2. Lighting and Shading

(a) Briefly describe *three* types of light and *four* types of light sources used in computer graphics. [7]

(b) Given a triangle mesh, briefly explain the *key* differences in how triangle normals are used to evaluate (i) *flat* shading, (ii) *Gouraud* shading, and (iii) *Phong* shading. [6]

(c) A triangle surface with vertices $\mathbf{p}_0$, $\mathbf{p}_1$, $\mathbf{p}_2$ is lit by a point light source with light intensity L and position $\mathbf{p}_L$. The viewer position is $\mathbf{p}_V$. Let $\mathbf{p}$ be centroid of the triangle. State the formula for Phong's reflection model to compute the light reflected from the point **p,** using $\mathbf{p}_0$, $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_L$, $\mathbf{p}_V$ and $\mathbf{p}$ to define any necessary intermediate quantities in the formula. It is assumed that the material properties of the surface are known, and you can use any symbols to represent the material properties in stating the formulae. [11]

Total: [24]

3. Camera Calibration

The relationship between a 3D point at world coordinates (x, y, z) and its corresponding 2D pixel at image coordinates (u, v) can be defined as a projective transformation using a 3×4 camera projection matrix P.

(a) Can the matrix P incorporate any *lens distortions* that might be in the camera? Briefly justify your answer. [4]

(b) Give *two* lists, one specifying the *intrinsic* camera parameters and the other giving the *extrinsic* camera parameters. [8]

(c) Show how P can be decomposed into a *product of matrices* that contain elements expressed in terms of the intrinsic and extrinsic camera parameters. [4]

(d) Give the *main steps* of an algorithm for computing the matrix P from a single image of a known 3D "calibration object." [8]

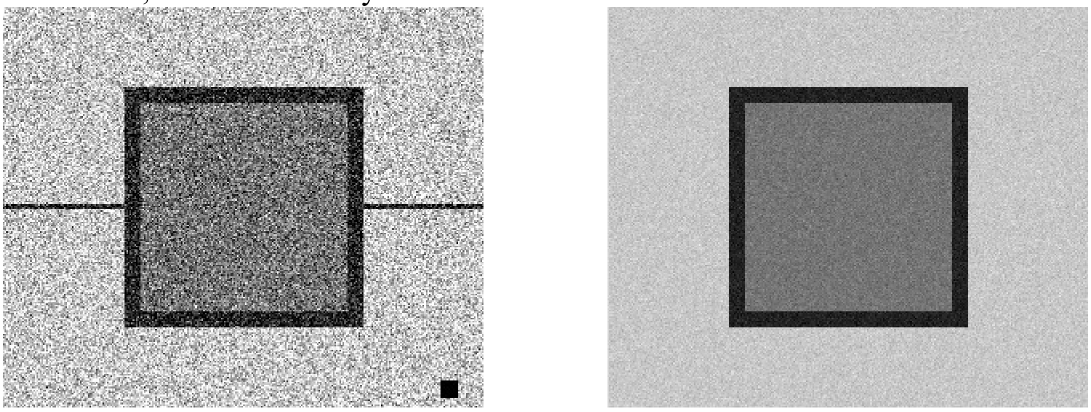
Total: [24]

4. Image Processing

(a) **[Image Smoothing]** In the figure below, when the image on the left was filtered using a *smoothing filter*, the result was the image on the right. The small black square on the lower right hand corner of the original image shows the size of the mask that was used. That small square is not part of the original image. In your opinion, which of the following filters most likely produced the image on the right?

1) box filter; 2) Gaussian low pass filter; 3) median filter.

For *each* of the *three* possible filters listed above, give *at least one* reason why you think it was, or was not, the filter actually used. [8]



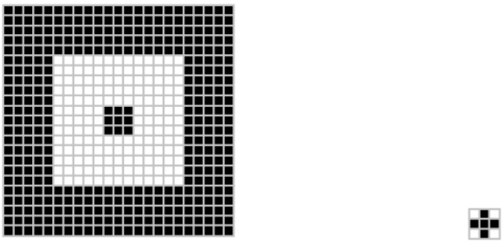
(a) original image (b) filtered image

(b) **[Edge Detection]** Show how an *approximation* to the first derivative of an image can be obtained by convolving the 1D image with the *kernel* [1 −1] where the image is defined as [55 63 82 97 110 121 126 128]

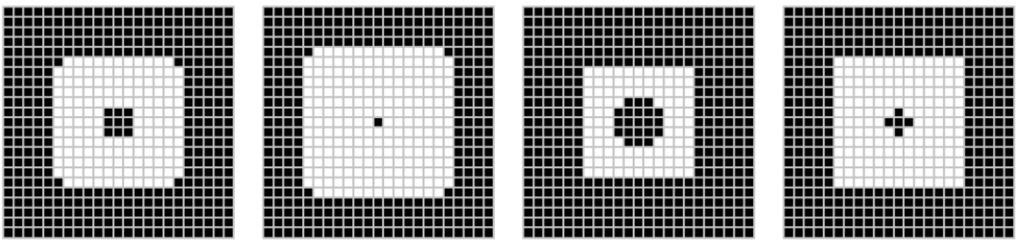
Ignore computing a value for the first and last image pixels (in other words, your result will comprise 6 values). In addition to showing the result of the convolution, indicate where edges would be detected and why. [8]

(c) **[Mathematical Morphology]** Given the original image and the structuring element (black for 1 and white for 0), describe *which* morphological operator (erode, dilate, open, close) has been used to create *each* of the 4 images (A-D) below. [8]

Total: [24]



Original image structuring element



A B C D

1. (a)

- The value of $\mathbf{a}\cdot\mathbf{b}$ is 0, because $|\mathbf{a}\cdot\mathbf{b}|=|\mathbf{a}|\cdot|\mathbf{b}|\cos\theta=3\cdot3\cdot0=0$, where θ is the angle between the **a** and **b**, which is 90°. [2]
- The length of $\mathbf{a}\times\mathbf{b}$ is 9, because $|\mathbf{a}\times\mathbf{b}|=|\mathbf{a}|\cdot|\mathbf{b}|\sin\theta=3\cdot3\cdot1=9$. [2]

(b) The Cartesian coordinate is (6/4, 2/4) = (1.5, 0.5). [3]

(c) Object, world, and camera reference frames. [3]

(d)

- Simple: edges of polygon do not intersect [2]
- Flat and convex [2]
- Flat: polygon lies in a plane [1]
- Convex: given any two points inside the polygon, the straight line segment. [1]

(e) C) is the image drawn by the code segment. [2]

- OpenGL performs transformation operations in a reverse order, i.e., the last command in the code executes first. [1]
- The transformations between glPushMatrix and glPopMatrix have no effect on the objects outside the range of the two commands. [1]
- Line 10 draws a sphere, which is translated in *y* direction by 2.0 (Line 9), and in *x* direction by -2.0 (Line 7), and then in *x* direction by 2.0 (Line 1), so its position is at (0.0, 2.0, 0.0) . [1]
- Line 8 draws a sphere, which is translated in *x* direction by -2.0 (Line 7), and then in *x* direction by 2.0 (Line 1), so its position is at (0.0, 0.0, 0.0) . [1]
- Line 5 draws a sphere, which is translated by (2.0, 2.0, 0.0) (Line 4), and then in *x* direction by 2.0 (Line 1), so its position is at (4.0, 2.0, 0.0) . [1]
- Line 2 draws a sphere, which is translated in *x* direction by 2.0 (Line 1), so its position is at (2.0, 0.0, 0.0) . [1]
- The above analyses show that (C) is the correct answer.

Total: [24]

2. (a)

- Ambient light: comes from all directions and is reflected in all directions; intensity is the same everywhere. [1]
- Diffuse light: comes from a specific direction and is reflected in all directions; intensity depends on the angle between the incoming light and the surface normal. [1]
- Specular light: comes from a specific direction and is reflected in a preferred direction; intensity depends on the angle between the direction of the perfectly reflected light and the direction to the viewer. [1]
- Ambient light source: light from the environment. [1]
- Directional light source: light from infinite distance in a specified direction. [1]
- Point light source: light from a single point. [1]

- Spot light source: light emitted in a cone. [1]

(b)

- In flat shading, the triangle normal is directly used to evaluate the reflected light intensity on the whole triangle. [2]
- In Gouraud shading, the triangle normal is combined with the normals of its neighbouring triangles to calculate its vertex normals. The vertex normal is then used to evaluate the reflected light intensity on the vertex, and the reflected light intensities on the whole triangle are calculated by bilinear interpolation of the reflected light intensities of its vertices. [2]
- In Phong shading, the triangle normal is also combined with the normals of its neighbouring triangles to calculate its vertex normals. But different from Gouraud shading, the vertex normals of the triangle are used to calculate the normals on each pixels of the whole triangle image through bilinear interpolation. And the reflected light intensity is evaluated on each pixels based on the obtained normals. [2]

(c)

- The normal of the triangle is calculated by $\mathbf{n}' = (\mathbf{p_1} - \mathbf{p_0}) \times (\mathbf{p_2} - \mathbf{p}_1)$, and $\mathbf{n} = \frac{\mathbf{n}'}{|\mathbf{n}'|}$. [3]
- The centroid of the triangle is $\mathbf{p} = (\mathbf{p_1} + \mathbf{p}_2 + \mathbf{p}_3)/3$ [1]
- The light direction is calculated by $\mathbf{l} = \frac{\mathbf{p}_L - \mathbf{p}}{|\mathbf{p}_L - \mathbf{p}|}$ [1]
- The viewer direction is calculated by $\mathbf{v} = \frac{\mathbf{p}_v - \mathbf{p}}{|\mathbf{p}_v - \mathbf{p}|}$ [1]
- The perfect reflection direction is calculated by $\mathbf{r} = 2(\mathbf{n} \cdot \mathbf{l})\mathbf{n} - \mathbf{l}$ [3]
- Let the material ambient, diffuse, and specular properties are R_a, $R_{d,}$ R_s, respectively, and let the shininess exponent is σ, then according to the Phong's reflection model, the reflected light from point **p** is

$$\boldsymbol{I} = \boldsymbol{R}_a \boldsymbol{L} + \boldsymbol{R}_d (\mathbf{n} \cdot \mathbf{l}) \boldsymbol{L} + \boldsymbol{R}_s (\mathbf{v} \cdot \mathbf{r})^{\sigma} \boldsymbol{L} \quad [2]$$

Total: [24]

3. (a) No, because lens distortions are usually highly nonlinear, which cannot be represented in P. [4]

(b)

- The five intrinsic parameters are the focal length, f, principal point coordinates, (u_0, v_0), and the pixel size scaling parameters, k_u, k_v. [4]
- The six extrinsic parameters are the three translational and three rotational parameters that define the rigid body motion between the world coordinate frame and the camera coordinate frame. [4]

(c) [4]

- $$\mathbf{P} = \mathbf{K}[\mathbf{R} \mid \mathbf{T}] = \begin{bmatrix} fk_u & 0 & u_0 \\ 0 & fk_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix}$$

(d)

- Since P is a 3 × 4 matrix with 11 unknowns (the overall scale of P does not matter), observing a 3D scene containing at least 6 known points (in a non-degenerate configuration) is sufficient. [2]
- Each point gives a pair of equations of the form
$$u = \frac{su}{s} = \frac{P_{11}X + P_{12}Y + P_{13}Z + P_{14}}{P_{31}X + P_{32}Y + P_{33}Z + P_{34}}, \quad v = \frac{sv}{s} = \frac{P_{21}X + P_{22}Y + P_{23}Z + P_{24}}{P_{31}X + P_{32}Y + P_{33}Z + P_{34}}$$ [2]
- Rewrite these 12 equations in the form A**p** = 0, where **p** is a 12×1 vector of unknowns, and A is a 12×12 matrix of coefficients. [2]
- Use least squares to solve for **p** by computing the eigenvector corresponding to the smallest eigenvalue of A^TA. Note: This is only an approximate solution and often this is then used as a starting point for a nonlinear optimization. [2]

Total: [24]

4. (a)

- 3) median filter [2]
- Cannot be the averaging filter since the edges are not blurred. [2]
- Cannot be the Gaussian filter because the edges are not blurred, either. [2]
- Must be the median filter, since it reduces noise, eliminates small objects and does not blur edges. [2]

(b)

- 8 19 15 13 11 5 [3]
- Intensity discontinuities occur at the maxima of the first derivative, which this approximates. [3]
- Here the maximum of 19 corresponds to the position where an edge is detected, corresponding to a position between the pixels with intensities 63 and 82 (or associated with one of these two pixels). [2]

(c) A: close, B: erode, C: dilate, D: open [8]

Total: [24]