IT575 Computational Shape Modeling End-Sem Examination

1$^{\text{st}}$ May 2017(5:30 pm) - 3$^{\text{rd}}$ May 2017 (09:00/09:30 am)

Maximum Marks - 45

NOTE:

1. Please remember that you are not allowed to discuss this paper with anyone, using any medium. Please include the following statement at the end of your answer sheet followed by your signature:
I have cited all material: book, notes, websites or others, used while answering this exam. I have not received any help in person for this exam. These answers (and codes) are my independent attempt at solving the exam.
2. All references and resources required are provided in the Exam folder, if you feel the urge to refer to more resources, you may use the library or the internet. Do cite your resources clearly.
3. Strictly follow the submission deadlines: Answer scripts: 3 May 2017 09:30 am, MATLAB codes: 3 May 2017, 09:00 am.
4. All plots should be included in a separate pdf file, ensure each plot refers to the right question number, and write your name and ID no on the pdf file. For this purpose, you may simply paste plots in a Word document and create a pdf file from it.
5. I will be available to answer your doubts/queries tomorrow (2 May 2017) 10:30 - 11:30 am in my office.

---

1. Let $S \subset \mathbb{R}^3$ be a smooth regular surface embedded in $\mathbb{R}^3$, with parameterization $\sigma : U \subset \mathbb{R}^2 \to S$, defined as

$$\sigma(u,v) = (u\cos v, u\sin v, u),\ \forall (u,v) \in (0,\infty) \times [0,2\pi).$$

Let $f : S \to \mathbb{R}$ be a scalar function defined by $f(q) = ||q-p||^2_{\mathbb{R}^3}, \forall q \in S$, where p is a fixed point in $\mathbb{R}^3 \setminus S$. Let $\tilde{f} = f \circ \sigma$ be the function f defined in local coordinates, i.e., $\tilde{f} : U \subset \mathbb{R}^2 \to \mathbb{R}$. [3+3+3+2+3 = 14]

(a) Minimizing the function f on S can be thought of as finding the projection of p to S. Find the projection for any point $p \in \mathbb{R}^3 \setminus S$ by analytically finding the minimizer (u^*, v^*) for $\tilde{f}$ and then computing the projection as $\sigma(u^*, v^*)$.

(b) The differential of $\tilde{f}$ at any local coordinate (u,v) is a linear map $D\tilde{f}_{(u,v)} : \mathbb{R}^2 \to \mathbb{R}$. Using chain rule, one also has $D\tilde{f}_{(u,v)} = Df_q \circ D\sigma_{(u,v)}$, where $q = \sigma(u,v)$, and Df and $D\sigma$ are differentials of f and σ respectively. Note that $D\sigma_{(u,v)} : \mathbb{R}^2 \to T_qS$ and $Df : T_qS \to \mathbb{R}$. Here we have used the fact that $T_{(u,v)}\mathbb{R}^2 \simeq \mathbb{R}^2$ and $T_{f(q)}\mathbb{R} \simeq \mathbb{R}$.

In the usual basis $b_1 = \{u, v\}$ for $\mathbb{R}^2$ and $b_2 = \{\sigma_u, \sigma_v\}$ for T_qS, one can see that the representation of the linear map $D\sigma$ is $[D\sigma]_{b_1}^{b_2} = I_2$, the 2×2 identity matrix. Thus $[D\tilde{f}] = [Df]$. Since the domain of Df is T_qS, this allows us to interpret $D\tilde{f}$ as a map from T_qS to $\mathbb{R}$. Moreover the relation between differential and gradient is $Df_q(w) = \langle \nabla f_q, w \rangle$, $\forall w \in T_qS$. Use these facts to geometrically interpret the condition for minima for the given f.

(c) Implement a MATLAB function `myproj.m` that takes in the point p, an arbitrary local coordinate (u_0, v_0), and does a gradient descent optimization of f. The MATLAB function should plot a few intermediate points obtained.

(d) Is this projection unique for all points $p \in \mathbb{R}^3 \setminus S$? Is the projection defined for every point in $\mathbb{R}^3 \setminus S$? Justify and show an example for each case.

(e) Let $\sigma_1 : U \subset \mathbb{R}^2 \to S$ and $\sigma_2 : W \subset \mathbb{R}^2 \to S$ be two parameterizations of the same smooth regular surface S. Does the minima of a function $f : S \to \mathbb{R}$ computed via functions defined on local coordinates, i.e., $\tilde{f}_1 = f \circ \sigma_1$ and $\tilde{f}_2 = f \circ \sigma_2$ differ? Prove either ways.

2. We say that two surfaces S_1 and S_2 are *locally isometric* if there exists a function $f : S_1 \to S_2$ such that the first fundamental form at any $p \in S_1$ is the same as the first fundamental form at $f(p) \in S_2$. One may use any parameterizations for S_1 and S_2.

[3+3+2 = 8]

(a) Prove that the right circular cone, $\sigma(u, v) = (u \cos v, u \sin v, u)$, $(u, v) \in (0, \infty) \times [0, 2\pi)$ is locally isometric to an appropriate subset of the plane.

(b) Use this fact to compute paths(possibly shortest) on the cone, by mapping straight lines on the plane to the cone, whenever possible. Write a MATLAB code `GeodesicCone.m` that takes in two points in $\mathbb{R}^3$, projects them to the cone, and generates the mapping of the straight line in local coordinates to the cone.

(c) Do you think this procedure will always produce minimal geodesics?

3. Pairwise geodesic distances for 900 points sampled on a surface are given in a matrix in the file `dMDS.mat`. Find out whether it is possible to isometrically embed the corresponding set of points in $\mathbb{R}^m$, for some positive integer m. Work out an approximate isometric embedding of these points in $\mathbb{R}^2$ using the SMACOF algorithm. [5]

4. Let $M = (V, F)$ be a given triangular mesh to be deformed, where V denotes the set of vertex coordinates and F encodes the triangular face indices. Let $V_h \subset V$ be the subset of handle vertices. As discussed in class, assuming the user provides the final position V_h' of the handle vertices V_h, the task of a shape deformation algorithm is to compute the final position of all non-handle vertices $V \setminus V_h$, such that the deformed mesh $M' = (V', F)$ appears natural.The deformation is achieved by translating every vertex $v \in V$ by a deformation vector d.

As derived in class, the deformation field over the continuous surface $\mathbf{d}$ that minimizes the energy functional

$$E(\mathbf{d}) = \int k_s(||\mathbf{d}_u||^2 + ||\mathbf{d}_v||^2) + k_b(||\mathbf{d}_{uu}||^2 + 2||\mathbf{d}_{uv}||^2 + ||\mathbf{d}_{vv}||^2)\, du\, dv,$$

satisfies the equation

$$\underbrace{(-k_sL + k_bL^2)}_{A}\, \mathbf{d} = 0, \tag{1}$$

where L is the Laplace-Beltrami operator, and k_s and k_b are positive reals specifying resistance to stretching and bending respectively. For a sampled surface represented as a mesh, one needs to compute the deformation vectors for each vertex in V. The collection of deformation vectors $\mathbf{d} \in \mathbb{R}^{|V|\times 3}$ satisfies Equation (1) where $L \in \mathbb{R}^{|V|\times|V|}$ denotes the area normalized Discrete LBO. The deformation vectors on the handles $v \in V_h$ are provided by the user as $d = v' - v = u$. Thus moving columns corresponding to the deformation vectors of the handles to the right hand side, and eliminating rows corresponding to the handle vertices from Equation (1), one can compute the deformation vectors at non-handle vertices, and thus compute the deformed mesh. For a detailed explanation refer [1]. [5+3 = 8]

(a) Write a MATLAB code `mymeshdeform.m` that takes as input mesh vertex coordinates $V \in \mathbb{R}^{|V|\times 3}$, triangle indices $F \in \mathbb{Z}^{m\times 3}$, where m denotes the number of triangles, a list of handle indices (moving followed by fixed) $I \in \mathbb{Z}^{|V_h|}$, and the corresponding deformation vectors $\mathbf{d}_h \in \mathbb{R}^{|V_h|\times 3}$, and outputs the vertices V' of the deformed mesh obtained by solving the above set of equations. Note that you should use the area normalized discrete LBO given in the file `computeLaplaceBeltrami.m`. An example of an input data your should try and run your code on in given in `cylinderdata2017.m`. Show a few plots depicting the effect of parameters k_s and k_b.

5. Continuing with the same problem area of mesh deformation, and same notations and model as the previous question, let $H \in \mathbb{R}^{|V|\times|V|}$ be the eigenvector matrix of the discrete LBO. For large meshes, one may have to solve a large system of linear equations to compute the deformation vectors. To simplify the system, one can compute only the first few (say k) low frequency components of the deformation vector. Let $\hat{\mathbf{d}} = H^T\mathbf{d}$ denote the coefficients of $\mathbf{d}$ in the eigenbasis H. Let $\hat{\mathbf{d}}_l = H_l^T\mathbf{d}$ represent the low frequency components to be estimated, where $H_l \in \mathbb{R}^{|V|\times k}$ is the eigenvector matrix corresponding to low frequencies.

   Substituting $L \simeq H_l\Lambda_l H_l^T$ in Equation (1) gives rise to the following constrained optimization problem:

$$(-k_s\Lambda_l + k_b\Lambda_l^2)\hat{\mathbf{d}}_l = 0, \tag{2}$$

   such that at any handle vertex, the user specifications are satisfied: $(H_l\hat{\mathbf{d}}_l)_h = u$. Let $B \in \mathbb{R}^{|V_h|\times|V|}$ be a matrix such that each constraint (handle) corresponds to a row, and for each row i, the entry B_{ij} is set to 1 if the i^{th} constraint is imposed on vertex i, else $B_{ij} = 0$. Thus all constraints (user inputs) can be written as

$$BH_l\hat{\mathbf{d}}_l = \mathbf{d}_h. \tag{3}$$

   One can combine and simultaneously solve the two linear systems given in Equations (2),(3), in the least square sense to compute the low frequency estimate $\hat{\mathbf{d}}_l$ of the deformation vectors. Using inverse transform H_l, it is easy to obtain the collection of deformation vectors $\mathbf{d} = H_l\hat{\mathbf{d}}_l$, that can be used to compute the deformed mesh V'. For a detailed explanation of a very *similar* algorithm, refer [2]. [5 + 5 = 10]

(a) Write a MATLAB code `myspectralmeshdeform.m` that takes the same set of inputs as described in the previuos question apart from the number of frequencies to be used, which outputs the vertices of the deformed mesh.

(b) Do you think the handle constraints will be satisfied? Justify? Will adding a scalar weight on both sides of Equation (3) help? Justify and demonstrate with a few examples. Try out working with the data of a cylinder provided in `cylinderdata2017.mat`.

References

[1] Mario Botsch and Olga Sorkine. On linear variational surface deformation methods. *IEEE Transactions on Visualization and Computer Graphics*, 14(1):213–230, January 2008.

[2] Guodong Rong, Yan Cao, and Xiaohu Guo. Spectral mesh deformation. *Visual Computing*, 24:787–796, 2008.