Module: **M6. 3D Vision** **Final exam**
Date: April 30, 2015
Teachers: Coloma Ballester, Josep Ramon Casas, Francesc Moreno, Gloria Haro, Javier Ruiz
Time: 2h

- Books, lecture notes, calculators, phones, etc. are not allowed.
- All sheets of paper should have your name.
- Answer each problem in a separate sheet of paper.
- All results should be demonstrated or justified.

Problem 1 *1 Point*

(a) *(0.25 points)* How do we represent a homography in the 2D projective space?

A 2D homography is represented by a 3×3 non-singular matrix.

(b) *(0.25 points)* How does a 2D homography act on points and lines?

Let H be a 2d homography.

A point $\mathbf{x} \in \mathbb{P}^2$ is transformed by $\mathbf{x}' = H\mathbf{x}$.

A line $\mathbf{l} \in \mathbb{P}^2$ is transformed by $\mathbf{l}' = H^{-T}\mathbf{l}$.

(c) *(0.5 points)* Enumerate the different situations where two images are related by a homography.
A homography relates two images:

- of the same plane in the 3D scene;
- taken with a camera rotating about its centre;
- taken with the same static camera varying its focal length;
- the whole scene is far away from the camera.

Problem 2 *1.75 Points*

(a) Consider two image views of a plane objet. Let $\mathbf{x}_i$ in $\mathbb{P}^2$, $i = 1, \ldots, n$, be a set of points on the first image and let $\mathbf{x}'_i$ in $\mathbb{P}^2$, $i = 1, \ldots, n$, be a set of points on the second image such as, in pairs, they correspond: $\mathbf{x}_i \longleftrightarrow \mathbf{x}'_i$, $\forall i = 1, \ldots, n$.

(i) *(0.25 points)* How many corresponding pairs of points in general position do you need to compute the 2D homography H such that $\mathbf{x}'_i = H\mathbf{x}_i$, $\forall i = 1, \ldots, n$? (Recall that general position means that no three points are collinear).

We need at least four points in general position to compute a transformation H of $\mathbb{P}^2$ ($n \geq 4$).

(ii) *(0.75 points)* Describe the Direct Linear Transformation (DLT) algorithm to compute H.

Let $\mathbf{x}_i = (x_i, y_i, w_i)$, $\mathbf{x}'_i = (x'_i, y'_i, w'_i)$. To fix ideas, let us assume that the points $\mathbf{x}_i$ and $\mathbf{x}'_i$ are measured in images so that $\mathbf{x}_i = (x_i, y_i, 1)$ and $\mathbf{x}'_i = (x'_i, y'_i, 1)$, (x_i, y_i) and (x'_i, y'_i) are pixel coordinates. We still keep the notation $\mathbf{x}_i = (x_i, y_i, w_i)$, $\mathbf{x}'_i = (x'_i, y'_i, w'_i)$ which is more general. Writing $H = \begin{pmatrix} \mathbf{h}_1^T \\ \mathbf{h}_2^T \\ \mathbf{h}_3^T \end{pmatrix}$, (that is, $\mathbf{h}_i^T$ is the i row of H), we obtain $\mathbf{x}'_i = H\mathbf{x}_i = \begin{pmatrix} \mathbf{h}_1^T\mathbf{x}_i \\ \mathbf{h}_2^T\mathbf{x}_i \\ \mathbf{h}_3^T\mathbf{x}_i \end{pmatrix}$. In homogeneous coordinates we have the equations

$$\frac{x'_i}{w'_i} = \frac{\mathbf{h}_1^T\mathbf{x}_i}{\mathbf{h}_3^T\mathbf{x}_i},$$

$$\frac{y'_i}{w'_i} = \frac{\mathbf{h}_2^T\mathbf{x}_i}{\mathbf{h}_3^T\mathbf{x}_i}.$$

That is,

$$x'_i\mathbf{h}_3^T\mathbf{x}_i - w'_i\mathbf{h}_1^T\mathbf{x}_i = 0,$$
$$y'_i\mathbf{h}_3^T\mathbf{x}_i - w'_i\mathbf{h}_2^T\mathbf{x}_i = 0.$$

Thus, each correspondence $\mathbf{x}_i \longleftrightarrow \mathbf{x}'_i$ produces two equations for the 9 unknowns $\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3$, They are

$$\begin{pmatrix} \mathbf{0}^T & -w'_i\mathbf{x}_i^T & y'_i\mathbf{x}_i^T \\ w'_i\mathbf{x}_i^T & \mathbf{0}^T & -x'_i\mathbf{x}_i^T \end{pmatrix} \begin{pmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \mathbf{h}_3 \end{pmatrix} = \mathbf{0} \in \mathbb{R}^2,$$

which can be written in matricial form as $\mathbf{A}_i\mathbf{h} = \mathbf{0}$. Now, if we have $n \geq 4$ correspondences $\mathbf{x}_i \longleftrightarrow \mathbf{x}'_i$ we have $2n \geq 8$ homogeneous equations. If $\mathbf{A}$ denotes the matrix obtained by $\mathbf{A} = \begin{pmatrix} \mathbf{A}_1 \\ \dots \\ \mathbf{A}_n \end{pmatrix}$, then, the system of equations writes

$$\mathbf{A}\mathbf{h} = \mathbf{0} \in \mathbf{R}^{2n}.$$

The vector $\mathbf{h}$ is in the null space of $\mathbf{A}$. To compute it, we can compute the SVD decomposition of $\mathbf{A}$, $\mathbf{A} = UDV^T$. Then $\mathbf{h}$ is the last column of V. It corresponds to the singular vector of modulus 1 associated to the smallest singular value of $\mathbf{A}$.

(b) *(0.75 points)* Consider an image of a 3D scene containing plane objects. Explain the method of affine rectification via the vanishing line.

First, we compute the line at infinity ℓ on the image which has a projective distortion. To this goal, we take two sets of two parallel lines, be it $\ell^a, \ell^b, \ell^c, \ell^d$, and we compute the vanishing point of each pair of parallel lines, $v^{ab} = \ell^a \times \ell^b$ and $v^{cd} = \ell^c \times \ell^d$. We consider that ℓ^a and ℓ^b are two parallel lines. ℓ^c and ℓ^d are another pair of parallel lines with direction different from ℓ^a. Then, from these two points, which are on the vanishing line $\ell = (l_1, l_2, l_3)$, compute the vanishing line as $\ell = v^{ab} \times v^{cd}$.

Finally, the projective transformation of $\mathbb{P}^2$ given by $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_1 & l_2 & l_3 \end{pmatrix}$ affinely rectifies the image.

Moreover, the family of projective transformations of $\mathbb{P}^2$ that map ℓ to $\ell_\infty = (0, 0, 1)^T$ can be written as

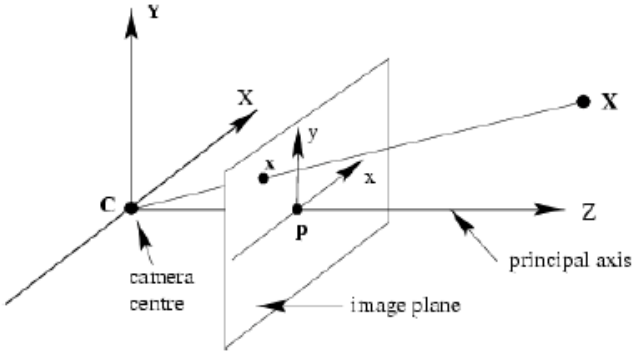
$$H_{a \leftarrow p} = H_a \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_1 & l_2 & l_3 \end{pmatrix},$$

where H_a is any affine transformation.

Problem 3 *0.75 Point*

(a) *(0.25 points)* Consider the pinhole camera model and consider a reference frame where its origin is the center of projection of the camera and the image plane is given by $Z = f$ where f is the focal length of the camera. Define camera centre, principal axis and principal point of the camera.

The camera centre (or optical centre) is the centre of projection.
The principal axis (or principal ray) is the line from the camera centre perpendicular to the image plane.
Principal point P: where the principal axis meets the image plane.



(b) *(0.5 points)* What is the general form of a finite projective camera matrix P ?

The general finite projective camera model is $\mathbf{x} = P\mathbf{X}$ with $\mathbf{X} \in \mathbb{P}^3, \mathbf{x} \in \mathbb{P}^2$ and

$$P = K[R|\mathbf{t}],$$

where $K = \begin{pmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{pmatrix}$ is the more general calibration matrix, $\mathbf{t} = -R\widetilde{\mathbf{C}}$, $\widetilde{\mathbf{C}}$ are the inhomogeneous coordinates of the camera centre $\mathbf{C}$ in the world coordinate frame, R and $\mathbf{t}$ are the rotation and the translation vector, respectively, that relate the inhomogeneous 3-vector $\widetilde{\mathbf{X}}$ representing the coordinates of a point in the world reference system with the same point in the camera coordinate frame, $\widetilde{\mathbf{X}}_{\text{cam}}$ (that is, $\widetilde{\mathbf{X}}_{\text{cam}} = R(\widetilde{\mathbf{X}}_{\text{w}} - \widetilde{\mathbf{C}})$, (p_x, p_y) are the coordinates of the principal point in a reference system in the image where the origin of coordinates in the image plane is situated at a corner of the image plane, α_x, α_y are parameters allowing the possibility of having non-square pixels for the image coordinates, and s is the skew parameter.

Problem 4 *0.5 Points*

Let H be a 3×3 homography that maps the points (u, v) of an image to points (x, y) on a 3D plane. Assuming that this matrix is known, and that we also know the 3×3 camera calibration matrix K, show how to compute the pose of the camera (rotation matrix R and translation vector t) with respect to the 3D plane.

Note: There is no need to show how to orthonormalize the matrix R. Assume we use a general routine $R_{ortho} = \texttt{orthonormalize}(R)$.

Let $H = \begin{bmatrix} h_1 & h_2 & h_3 \end{bmatrix}$ be the 3 columns of H. We have that:

$$\begin{bmatrix} u \cdot w \\ v \cdot w \\ w \end{bmatrix} = \begin{bmatrix} h_1 & h_2 & h_3 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{1}$$

We can write this homography in terms of K, R and t. For this we consider a 3D reference system on the plane, and write the coordinates of a point on it by $(x, y, 0)$, with null z component. The perspective projection of this point onto the image plane can be written as

$$\begin{bmatrix} u \cdot w \\ v \cdot w \\ w \end{bmatrix} = K \begin{bmatrix} R & t \end{bmatrix} \begin{bmatrix} x \\ y \\ 0 \\ 1 \end{bmatrix} = K \begin{bmatrix} r_1 & r_2 & r_3 & t \end{bmatrix} \begin{bmatrix} x \\ y \\ 0 \\ 1 \end{bmatrix}$$
$$= K \begin{bmatrix} r_1 & r_2 & t \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{2}$$

where $R = \begin{bmatrix} r_1 & r_2 & r_3 \end{bmatrix}$ are the 3 columns of R.

From direct comparison of Eq. 1 and Eq. 2 we have that $H = K \begin{bmatrix} r_1 & r_2 & t \end{bmatrix}$. If we write $G = \begin{bmatrix} g_1 & g_2 & g_3 \end{bmatrix} = K^{-1}H$ we can finally retrieve R and t as:

$$\begin{aligned} r_1 &= g_1 \\ r_2 &= g_2 \\ t &= g_3 \\ r_3 &= r_1 \times r_2 \end{aligned} \tag{3}$$

Since there is no guarantee that $R = \begin{bmatrix} r_1 & r_2 & r_3 \end{bmatrix}$ will be orthonormal, we finally apply:

$$R_{ortho} = \texttt{orthonormalize}(R)$$

Problem 5 *2 points*

We plan to estimate the fundamental matrix F between two images I and I' taken by the same camera using the 8-point algorithm.

(a) *(0.5 points)* Given two correspondences $p_1 = (10, 5), p_1' = (12, 5)$, and $p_2 = (20, 30), p_2' = (25, 30)$ write the first 2 rows of the matrix W that allows us to estimate the fundamental matrix F (expressed as a vector column f) with a homogeneous system ($Wf = 0$).

$$W = \begin{bmatrix} 10 \cdot 12 & 5 \cdot 12 & 12 & 10 \cdot 5 & 5 \cdot 5 & 5 & 10 & 5 & 1 \\ 20 \cdot 25 & 30 \cdot 25 & 25 & 20 \cdot 30 & 30 \cdot 30 & 30 & 20 & 30 & 1 \\ & & & \cdots & & & & & \end{bmatrix}$$

If the singular value decomposition of the matrix W can be expressed as:

$$W = UD \begin{bmatrix} 1 & 0 & -1 & 1 & -1 & 0 & -1 & 1 & 1 \\ 0 & 0 & 0 & 1 & -1 & 0 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 & -1 & 1 & 1 & -1 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ -1 & 1 & 0 & 0 & 0 & -1 & 1 & -1 & -1 \\ 1 & 0 & -1 & -1 & 1 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 1 & 1 & 1 \\ -1 & 1 & 0 & -1 & 1 & 1 & 1 & -1 & 0 \end{bmatrix}^T$$

(b) *(0.25 points)* Obtain a first approximation of the fundamental matrix.

$$F = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

Suppose now that the singular value decomposition of the fundamental matrix obtained in the previous question is

$$F = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}^T$$

(c) *(0.25 points)* Obtain a second approximation of the fundamental matrix that ensures all the epipolar lines cross at the same point (epipole).

$$F = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

(d) *(0.5 points)* Taken into account the fundamental matrix obtained above, are the two images I and I' rectified?

Yes as it only has two singular values different than 0

(e) *(0.5 points)* Enumerate three main disadvantages (or problems) with the 8-point algorithm and briefly comment a possible solution to them.

Sensitive to noise → Add more points to the estimation
Sensitive to false correspondence → RANSAC
Highly unstable → Normalize input data

Problem 6 *1 Point*

(a) *(0.25 points)* Describe the triangulation problem, i.e. what are the unknowns and the known data.

The triangulation problem is the problem of determining a point in 3D space given its projections onto two, or more, images.

The known data are:

- the projection matrices of the cameras,
- the point projections and their correspondences.

(b) *(0.5 points)* Which is the energy that the geometric triangulation method minimizes and how do we use its solution to solve the triangulation problem? Limit the problem to the two-view case and define every variable that you use in the energy.

Let $\mathbf{x}, \mathbf{x}' \in \mathbb{P}^2$ be the 2D projection points in each of the two views. Let F be the fundamental matrix that relates the two views, and [.] the projector operator that converts from homogeneous to cartesian coordinates. Then, the geometric triangulation problem is:

$$\min_{\widehat{\mathbf{x}},\widehat{\mathbf{x}}'} d^2(\mathbf{x},\widehat{\mathbf{x}}) + d^2(\mathbf{x}',\widehat{\mathbf{x}}') = \min_{\widehat{\mathbf{x}},\widehat{\mathbf{x}}'} \|[\mathbf{x}] - [\widehat{\mathbf{x}}]\|_2^2 + \|[\mathbf{x}'] - [\widehat{\mathbf{x}}']\|_2^2$$

$$\text{such that } \widehat{\mathbf{x}}'^T F \widehat{\mathbf{x}} = 0.$$

Then, we get the 3D point $\widehat{\mathbf{X}}$ from $\widehat{\mathbf{x}}$ and $\widehat{\mathbf{x}}'$ (since the visual rays intersect).

(c) *(0.25 points)* Describe the main steps of a local algorithm for stereo matching given a pair of rectified images.

Take the left image as the reference image. Then, for every pixel in the left image

(i) Slide a window along the same line in the right image and compare its content to that of the reference window in the left image.

(ii) Pick pixel with minimum matching cost. This pixel determines the disparity.

Problem 7 *1 Point*

Consider the factorization method of Sturm and Triggs 1996 and assume we have three points seen by two cameras.

(a) *(0.5 points)* Write the measurement matrix and define its elements. Show how it relates to the 3D points and the camera projection matrices.

Let $\mathbf{X}_j \in \mathbb{P}^3$, $j = 1, 2, 3$, denote the 3D points, P^1 and P^2 the two camera matrices, and $\mathbf{x}_j^i \in \mathbb{P}^2$, $i = 1, 2$, the projected points.

From the projective equations $\mathbf{x}_j^i \equiv P^i \mathbf{X}_j$ we write $\lambda_j^i \mathbf{x}_j^i = P^i \mathbf{X}_j$, where λ_j^i are unknown scalar factors (projective depths).

We collect all projective equations into a matrix equation:

$$\begin{bmatrix} \lambda_1^1 \mathbf{x}_1^1 & \lambda_2^1 \mathbf{x}_2^1 & \lambda_3^1 \mathbf{x}_3^1 \\ \lambda_1^2 \mathbf{x}_1^2 & \lambda_2^2 \mathbf{x}_2^2 & \lambda_3^2 \mathbf{x}_3^2 \end{bmatrix} = M = \begin{bmatrix} P^1 \\ P^2 \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \mathbf{X}_3 \end{bmatrix},$$

where M is the measurement matrix.

(b) *(0.5 points)* How do we extract a projective reconstruction once we have the Singular Value Decomposition of the measurement matrix?

Using the SVD of M and the fact that M is at most of rank 4:

$$M = \underbrace{U D_4}_{6 \times 4} \underbrace{V_4^T}_{4 \times 3}$$

we have

$$\begin{bmatrix} P^1 \\ P^2 \end{bmatrix} = U D_4, \qquad \begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \mathbf{X}_3 \end{bmatrix} = V_4^T.$$

Problem 8 *0.5 Points*

Describe how to estimate the image of the absolute conic given three vanishing points from three orthogonal scene directions and assuming that the camera has zero skew and square pixels.

We have five constraints on the image of the absolute conic ω:

$$\mathbf{u}^T \omega \mathbf{v} = 0$$

$$\mathbf{u}^T \omega \mathbf{z} = 0$$

$$\mathbf{v}^T \omega \mathbf{z} = 0$$

$$\omega_{11} = \omega_{22}$$

$$\omega_{12} = 0$$

where $\mathbf{u} = (u_1, u_2, u_3)^T$, $\mathbf{v} = (v_1, v_2, v_3)^T$, and $\mathbf{z} = (z_1, z_2, z_3)^T$ are the image of the vanishing points in homogeneous coordinates.

In matrix form:

$$A\omega_V = \mathbf{0},$$

where $\omega_V = (\omega_{11}, \omega_{12}, \omega_{13}, \omega_{22}, \omega_{23}, \omega_{33})^T$ and

$$A = \begin{pmatrix} u_1v_1 & u_1v_2 + u_2v_1 & u_1v_3 + u_3v_1 & u_2v_2 & u_2v_3 + u_3v_2 & u_3v_3 \\ u_1z_1 & u_1z_2 + u_2z_1 & u_1z_3 + u_3z_1 & u_2z_2 & u_2z_3 + u_3z_2 & u_3z_3 \\ v_1z_1 & v_1z_2 + v_2z_1 & v_1z_3 + v_3z_1 & v_2z_2 & v_2z_3 + v_3z_2 & v_3z_3 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \end{pmatrix}.$$

The solution ω_V is the null vector of A.

Problem 9 *1 Point*

3D sensors

Rendering is the process of synthetically generating a raster image from a 3D model of a scene. Rendering, usually implemented in a graphics pipeline, tries to mimic the equivalent physical process of image acquisition through an RGB capture device. The synthesis is *computed* from the information in the scene model about (1) scene geometry, (2) camera viewpoint, (3) texture, (4) lighting and (5) shading.

In general, and only from the raster image resulting of a rendering process, it is not possible to recover the 5 items of information listed above. However, RGB+Depth (RGBD) capture devices perform an operation which, in some way, can be considered inverse to the rendering operation in Computer Graphics. Let's analyze this statement more closely.

(a) Could you provide a generic specification of the exact information recovered in an RGBD sensor?

Typical information recovered by RGBD sensors consists of a sequence of raster images with the following characteristics:

- WxH pixels
- (typ) 30fps
- 24 bits RGB values per pixel
- 16 bits D value per pixel, indicating distance in the Z axis (with some scale factor and boundaries, and allowing NaN values for pixels for which the sensor has been unable to recover the Z coordinate).

(b) In commercial depth sensors (Kinect, Asus Xtion), how is the D component of this information recovered?

By means of a light coding system, which projects a light pattern into the scene and estimates the disparity of this pattern for every horizontal point in a scan line by active triangulation.

(c) How can we recover scene geometry (1) from the RGBD data? Would we do we need additionally?

We may need intrinsic camera calibration parameters to convert (x,y) coordinates of the pixel position to actual X,Y coordinates in 3D space at the Z (Depth) position delivered by the sensor. In this way, we can recover the scene geometry of the visible scene surfaces for which we have a reliable depth estimate from the sensor.

(d) Is it possible to recover camera viewpoint (2) from RGBD data?

We can recover the camera viewpoint coordinates in the camera centered coordinate system. We would need either extrinsic camera parameters (or some information about the scene structure) to be able to obtain the camera viewpoint coordinates in the world coordinate system.

(e) Is there any way to recover reflectivity and lighting properties (3, 4, and 5) from RGBD data?

No. All these parameters from the scene (texture, lighting and shading) combine to reflect a certain amount of intensity captured by the sensor at a given point.

(f) Reason whether it would be possible to recover (3, 4 and 5) with any other imaging sensor device (active or passive, even if based in other imaging wavelengths)

No, or at least, not easily if the only information captured by the sensor is reflected light (in any wavelength).

Problem 10 *0,5 Points*

Meshing

We discussed the advantages of the new trend of directly processing the 3D raw data (point clouds) produced by 3D scanners. For image/scene analysis applications, what are the advantages and disadvantages of this strategy regarding actual XYZ values and connectivity when we compare it with working with a meshed input? Otherwise, what would be the main motivation for meshing?

The advantages of working directly with raw point clouds are related to working with actual XYZ values for the data captured by the sensor. Any processing to obtain a mesh, may remove noise, but could also smooth relevant variations of surface geometry. The disadvantages are related to the absence of connectivity information in a pointcloud. Any estimated connectivity would be just the result of an estimation process.