Introduction to Computer Vision S22 Assignment #1

Due March 23/(Wed)

1. 2D Fourier Transform of Images

- Load images *img1*, and *img2* with ‘`imread`’;
- Display the images.
- Next resize the img2 to be the same size as the img1 with `resizedCheckerboard = imresize(img2, size(img1));`.

- Perform 2D DFT on *img1*, and *img2*. (with ‘`fft2'` and ‘`fftshift'`)
- Display the magnitude and phase of each image, *img1_mag, img1_phase, img2_mag,* and *img2_phase*. (use ‘`mat2gray'`. To display magnitude you need to rescale it first, like `Y=log(abs(X))`
- Synthesize new images *new_img1* and *new_img2* by combining *img1_mag* + *img2_phase*, and *img2_mag* + *img1_phase*, respectively.
- Discuss and analyze the results you obtained.

2. Perspective Image Transforms

- A 2D projective transform is a projection defined by a 3x3 matrix **M** which transforms a point [*x y*] into [*u v*] where [*p q r*]^T = **M** [*x y 1*]^T, and *u = p/r* and *v = q/r*. (Look up homogeneous coordinates for more information on why you add the extra 1.)

- Perform the 2D projective transforms defined by the following 4 matrices for *img1*, and save the results as *projected1.png* through *projected4.png.* (Following matrices are in the forms of **M**^{T})

1.6322	0	0	1.4219	0.3183	0.0013
0.2120	1.6336	0.0013	0	1.4206	0
-101.9757	-0.6322	1.0000	-0.4206	-101.8704	1.0000
0.7033	-0:2239	-0:0009	1.1044	-0.3493	0.0003
0	0.9991	0	0.0011	1.5066	0.0011
0.2958	0.2239	1.0000	-0.1041	-0.1560	1.0000

3. Camera Calibration

- A camera projection matric **P** maps a world point $\mathbf{X}_i = [X_i \quad Y_i \quad Z_i \quad 1]^T$ to its corresponding image point $\mathbf{x}_i = \omega[u_i \quad v_i \quad 1]^T$ such that $\mathbf{x}_i = \mathbf{P}\mathbf{X}_i$, which gives two equations w.r.t. the unknown parameters of $\mathbf{P} = [m_{ij}]$:

$$m_{11}X_i + m_{12}Y_i + m_{13}Z_i + m_{14} - m_{31}u_iX_i - m_{32}u_iY_i - m_{33}u_iZ_i - m_{34}u_i = 0$$
$$m_{21}X_i + m_{22}Y_i + m_{23}Z_i + m_{24} - m_{31}v_iX_i - m_{32}v_iY_i - m_{33}v_iZ_i - m_{34}v_i = 0$$

- Given $n \geq 6$ pairs of corresponding points, we have an over-determined system of equations, $\mathbf{Ap} = \mathbf{0}$, where **p** is the vectorized form of **P**.
- Consider the SVD of **A** as $\mathbf{A} = \mathbf{UDV}^T$. Then the least square solution $\mathbf{p} = min_{\mathbf{p}}||\mathbf{Ap}||$, $||\mathbf{p}||$=1 is the last column of $\mathbf{V}$, which is the eigenvector corresponding to the smallest eigenvalue. And

then you can construct the camera projection matrix **P** up to scale.

a) Show that the solution **p** is the last column of **V** correspoding to the smallest eigenvalue of **A**.

b) Given the following corresponding points, determine the camera projection matric **P** using the SVD method.

(u	v)	(X	Y	Z)
880	214	312.747	309.140	30.086
43	203	305.796	311.649	30.356
270	197	307.694	312.358	30.418
886	347	310.149	307.186	29.298
745	302	311.937	310.105	29.216
943	128	311.202	307.572	30.682
476	590	307.106	306.876	28.660
419	214	309.317	312.490	30.230
317	335	307.435	310.151	29.318
783	521	308.253	306.300	28.881
235	427	306.650	309.301	28.905
665	429	308.069	306.831	29.189
655	362	309.671	308.834	29.029
427	333	308.255	309.955	29.267
412	415	307.546	308.613	28.963
746	351	311.036	309.206	28.913
434	415	307.518	308.175	29.069
525	234	309.950	311.262	29.990
716	308	312.160	310.772	29.080
602	187	311.988	312.709	30.514

c) Another method to determine the **P** is using the pseudo inverse. By setting m_{34}=1, we have $\mathbf{Ap} = \mathbf{b}$, and the least square solution $\mathbf{p} = min_{\mathbf{p}}\|\mathbf{Ap} - \mathbf{b}\|$, $\|\mathbf{p}\|$=1 become $\mathbf{p} = \mathbf{A}^{+}\mathbf{b}$, where $\mathbf{A}^{+} = (\mathbf{A}^{T}\mathbf{A})^{-1}\mathbf{A}^{T}$ is the pseudo inverse of **A**. Determine **P** for the data in b) using a pseudo inverse method.

Implementation & Submission instructions:

- Implementation instruction: Use Python for your implementation.
- Submission instructions:
- Upload the electronic file that includes the report, source code, and data in a single zip format with the name “**ICV_assignment#1_yourname.zip**” on the ETL class homepage.
- The report should include the brief description of the problems, code, results, and discussions

- Reference
  - Quick tutorial page on how to use Matlab
  - Quick tutorial page on how to use Image Processing Toolbox
  - OpenCV
  - OpenCV-Python Tutorials

Note: All works should be individual-based. NO copy is allowed.