Assignment 4
Image Restoration Applications Are COOL!

Homeworks Guidelines and Policies

- ***What you must hand in.*** It is expected that the students submit an assignment report (HW4_[student_id].pdf) as well as required source codes (.m or .py) into an archive file (HW4_[student_id].zip).
- ***Pay attention to problem types.*** Some problems are required to be solved *by hand* (shown by the icon), and some need to be implemented (shown by the icon).
  Please don't use implementation tools when it is asked to solve the problem by hand, otherwise you'll be penalized and lose some points.
- ***Don't bother typing!*** You are free to solve by-hand problems on a paper and include picture of them in your report. Here, cleanness and readability are of high importance. Images should also have appropriate quality.
- ***Reports are critical.*** Your work will be evaluated mostly by the quality of your report. Don't forget to explain what you have done, and provide enough discussions when it's needed.
- ***Appearance matters!*** In each homework, 5 points (out of a possible 100) belongs to compactness, expressiveness and neatness of your report and codes.
- ***Python is also allowable.*** By default, we assume you implement your codes in MATLAB. If you're using Python, you have to use equivalent functions when it is asked to use specific MATLAB functions.
- ***Be neat and tidy!*** Your codes must be separated for each question, and for each part. For example, you have to create a separate .m file for part b. of question 3. Please name it like p3b.m.
- ***Use bonus points to improve your score.*** Problems with bonus points are marked by the icon. These problems usually include uncovered related topics or those that are only mentioned briefly in the class.
- ***Moodle access is essential.*** Make sure you have access to Moodle because that's where all assignments as well as course announcements are posted on. Homework submissions are also done through Moodle.

- ***Assignment Deadline.*** Please submit your work **before the end of July 10th**.
- ***Delay policy.*** During the semester, students are given 7 free late days which they can use them in their own ways. Afterwards there will be a 25% penalty for every late day, and no more than three late days will be accepted.
- ***Collaboration policy.*** We encourage students to work together, share their findings and utilize all the resources available. However you are not allowed to share codes/answers or use works from the past semesters. Violators will receive a zero for that particular problem.
- ***Any questions?*** If there is any question, please don't hesitate to contact me through the following email address: **ali.the.special@gmail.com**.

1. When the Shah of Iran Took the Pictures (16 Pts.)

Keywords: *Image Denoising, Spatial Filtering, Additive Noise, Gaussian Noise, Impulse (Salt-and-Pepper) Noise, Periodic Noise*

Although a weak and incapable king, Naser al-Din Shah was a talented artist. Not only was he an expert in pen and ink drawing, he was also an excellent photographer (Figure 1). He brought the first camera to Iran, and is known as one of the first Iranian photographers in the history. He was also responsible for establishing a photography studio in Golestan Palace.

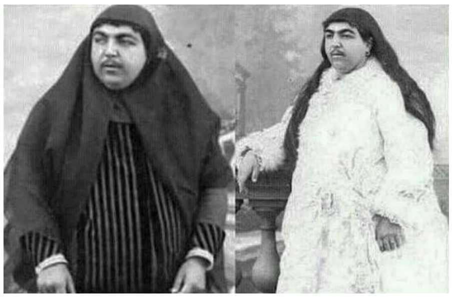
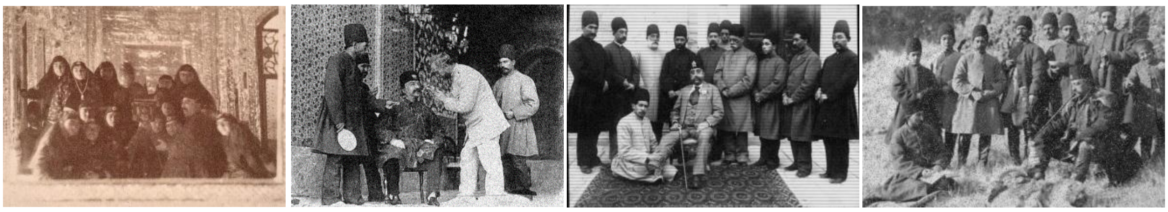


Figure 1 Two photos taken by the Shah himself, showing princess Anis al-Doleh, which is said to be the prettiest girl of her time. History claims 13 men committed suicide because she rejected them!

The Shah was mainly interested in recording private, daily life of his household, including his 85 wives in his harem. Although his enthusiasm lead him to create a precious collection of documents of Iranian lifestyle in Qajar era, most of the remaining photos are noisy due to the rudimentary methods used in the process of recording them.

Four different images of that era are given, each with the presence of different types of noise. Your task is to first detect the noise type and then to implement a method to reduce their effects.



(a) **(b)** **(c)** **(d)**

Figure 2 Different photos of Nasir al-Din Shah, corrupted with different types of noises (a) King in his harem, taking mirror selfie with his wives (b) King receiving dental treatment (c) King receiving a service known as "Pacheh-khaari" (d) King smoking hookah after hunting a ram

Note: All noise reduction methods must be implemented. In other words, no built-in function and library is allowed to be used in the process of noise removal, except for routine operations.

2. It Don't Matter If You're Trump or Biden! (24 Pts.)

Keywords: *Image Morphing, Facial Morphing, Facial Features, Cross Dissolve Method, Delaunay Triangulation, Affine Transformation*

On November 1991, Michael Jackson released a music video named *Black or White*, which was based on his single from the album *Dangerous*. Similar to most of his music videos, Black or White soon became hugely popular worldwide, with an audience of 500 million views, the world record for most views for a music video. As well as a humorous storyline, the music video was considered a cinematic masterpiece at the time, with several innovative visual techniques which hadn't been seen before.

Figure 3 In the last minute of the music video, several people from different races dance as their face change into another, while repeatedly singing "black or white".

At the end of the music video, several people of different ethnicities and nationalities (Figure 3) dance as they *morph* into one another, emphasizing the message of the song which was basically about the concept of racial equality (The main verse was *"It don't matter if you're black or white"*). The technique, which hadn't been previously used elsewhere, is today known as **Image Morphing**, and is often used as a fun way to convert one face (or sometimes object) into another, Figure 4.

Figure 4 From Trump to Biden. Face morphing is often used in media to convert a celebrity face into another

Image Morphing can be performed through various approaches. Here, we first introduce a naïve way to handle the problem, and then define an algorithm based on **Geometric Transformation**, which successfully overcomes the first method's deficiency.

I. Naïve Method

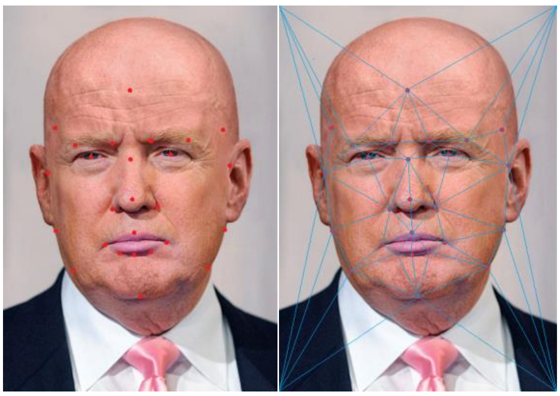
The first idea which may come to the mind is to simply interpolate whole images. Assuming M as the initial image and N as the target image, each pixel in the morphed image I can be interpolated as follows:

$$I(i,j)=(1-\alpha)\cdot M(i,j)+\alpha\cdot N(i,j),\quad 0\le\alpha\le 1$$

where α controls the gradual transformation from M to N. This technique is known as *cross-dissolve* in the film industry, and is able to balance coloration in the middle frames. However, it leaves a "ghosting" effects in the intermediate images, especially if the two faces aren't perfectly aligned.

II. Triangulation Method

In the previous method, only the "photometric" side of the transformation was addressed, and the "geometric" side was ignored. We therefore need to find a one-to-one correspondence between certain parts of the two images (e.g. lips, eyes, nose, etc.), so as to know which part of the initial image should be morphed into which part of the target image. This can be done through a new algorithm, which is defined in the following steps:



(a) (b)

Figure 5 The process of face triangulation (a) Specific landmarks are detected, denoting important regions of the face (b) Triangulation algorithm is performed

i. Finding corresponding points. First, a set of corresponding keypoints must be extracted from each images. These keypoints play a major role in highlighting prominent facial features so as the interpolation becomes more specific. Here, we perform this step manually, using the reference image shown in Figure 5. In this image, 26 facial feature points are highlighted with red dots. In order to include the background into the morphing frames, at least 4 points (corners) must also be selected. Let $P_M=\{p_M^1,p_M^2,\dots,p_M^k\}$ and $P_N=\{p_N^1,p_N^2,\dots,p_N^k\}$ be the two set of points extracted from images M and N, respectively.

ii. Delaunay Triangulation. Next, a weighted mean of the points in the two sets must be calculated. More precisely, using P_M and P_N, we obtain another set of points $P_I = \{p_I^1, p_I^2, \dots, p_I^k\}$ in which

$$p_I^j = (1-\alpha) p_M^j + \alpha p_N^j$$

These points are the corresponding features in the intermediate image, and will be used later to apply the triangulation to the set P_I.

Triangulation operation divides the plane of the image into small triangles (Figure 5), using the points of the set P_I as vertices. We opt to use *Delaunay Triangulation*, since it does not produce overly skinny triangles. The triangulation should be performed for all the sets P_M, P_N and P_I, giving a one-to-one correspondence between triangles in the images M, N and I. We now have three sets of corresponding triangles, $T_M = \{t_M^1, t_M^2, \dots, t_M^l\}$, $T_N = \{t_N^1, t_N^2, \dots, t_N^l\}$ and $T_I = \{t_I^1, t_I^2, \dots, t_I^l\}$.

iii. Affine transformation. Now, we select a triangle t_M^j from the image M as well as its corresponding triangle t_I^j from the image I, and calculate the affine transform which converts the triangle t_M^j to t_I^j. Similarly, we calculate the transformation matrix of the corresponding triangle t_N^j in the image N to the triangle t_I^j. Then we apply the obtained transformations to all the triangles of both images M and N, and obtain warped images M' and N'. Finally, we calculate the intermediate image I by using the same equation in the naïve method:

$$I(i,j) = (1-\alpha)\cdot M'(i,j) + \alpha \cdot N'(i,j), \quad 0 \le \alpha \le 1$$

Hence, a series of intermediate images based on different values of α will be obtained. When shown in rapid succession, these images gives the effect of one image being converted into the other.

Two images 'rouhani.png' and 'raisi.png' are given. Consider the second as the target image.

(a) **(b)**

Figure 6 Input images given for this task (a) Initial image (b) Target image

a. Use the naïve method to convert the initial image to the target one. Calculate and display 10 intermediate images. Comment on the results.
b. Manually select keypoints in two images. Display the detected points.
c. Apply triangulation to both images, and display the results.
d. Calculate the transformations, and display 10 intermediate images. Compare the results with part (a).
e. Generate a two-second animation containing 61 frames. Frame 0 must be identical to the initial image, and frame 60 must be identical to the target image. In the video, each frame will be displayed for 1/30 of a second.

Note 1: The transition between frames should be smooth and the intermediate frames must be as realistic as possible.

Note 2: There's no restrictions on the usage of built-in functions and libraries.

Recommended MATLAB functions: `cpselect()`, `ginput()`, `delaunay()`, `imtransform()`, `cp2tform()`, `maketform()`, `tsearch()`, `interp2()`, `VideoWriter()`, `writeVideo()`

3. Getting to Know Google Earth Post-Processing Operations (25 Pts.)

Keywords: *Image Stitching, Panorama, Feature Extraction, Feature Matching, Geometric Transformation, Image Blending, Scale-Invariant Feature Transform (SIFT), Random Sample Consensus (RANSAC)*

Google Earth is an amazing tool which allows us to zoom around the world and explore a satellite view of our neighborhood, see 3D imagery of well-known landmarks and navigate through famous streets in big cities, all from the comfort of our homes. The imagery and data used by Google Earth is raw images taken by satellites and aircrafts and is collected through Google's partnerships with NASA, National Geographic and others. But what process converts these raw images into good-looking seamless maps we see?

Google receives billions of images taken in various conditions. After enhancing these images (e.g. noise reduction, haze removal, contrast and brightness adjustment, etc.), the arduous task is to link these images together. To accomplish this, Google relies on a simple, yet very effective algorithm, known as **Image Stitching**, the same technique which your smartphone uses when you want to take a panorama photo.

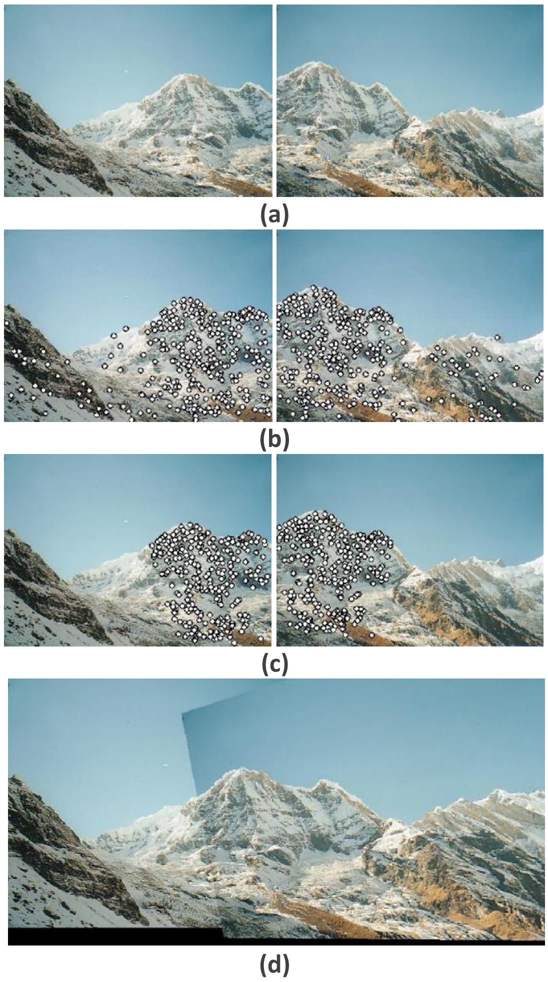


Figure 7 The process of image stitching (a) Initial images (b) Keypoints are detected in both images (c) A set of matched keypoints between two images are obtained, from which the geometric transformation between the two images is estimated (d) After applying the geometric transformation, images are blended and create the final panorama result

The process of image stitching can be divided into four steps:

I. Feature Detection. First, a set of interest points are detected in the images. Here, we consider the *Scale-Invariant Feature Transform (SIFT)* detector.

II. Feature Matching. Next, similar keypoints between the pair of images are detected. To perform this, a distance measure (e.g. Euclidean distance) between feature vectors must be considered. Then a threshold must be used to determine whether a pair of keypoints from two images shall be considered as a match.

III. Image Registration. The goal of this step is to use the corresponding points in the reference image and the target image to estimate a geometric relationship between the two images. For this purpose, we consider *RANdom SAmple Consensus (RANSAC)* algorithm which attempts to find a transformation which is less affected by mismatched points.

IV. Image Blending. Finally, the transformed image(s) must be combined by averaging the pixel values in the overlapped regions from two or more images. Another way to implement this step is to apply weighted averaging of overlapping pixel values where the weights vary linearly according to the distance of a pixel in the overlapped region to the centre of either one of the images.

In this problem, we aim to investigate the usage of this technique in three different features of Google Earth.

I. Satellite View

From its initial release in 2005, Google Earth has provided satellite views of earth varying in quality and resolution. Imagery of a certain area may be a combination of various satellite images, each taken from different angles, scales and positions. Hence, they are required to be stitched to make one integrated image of the area with no visible border, Figure 8.

(a) **(b)** **(c)**

Figure 8 An example of stitching two satellite images (a) First image (b) Another image of the same scene, with minor change in camera position (c) Resultant image

a. Three satellite images of our university are provided. Although they share similar scales, the camera has been slightly rotated when capturing these images. Use the procedure described above to obtain an integrated satellite image of our university campus.

II. 3D Aerial View

In 2012, Google started to deploy small, camera-equipped airplanes to fly above various cities and take pictures of their buildings. The goal was to utilize these images and create 3D maps of those areas. Today, Google Earth 3D imagery has been expanded to over 60 countries, including every U.S. state. By allowing the users to view photorealistic 3D imagery of buildings (and even objects like cars, trees and billboards), this feature has brought a whole new experience to Google Earth users, Figure 9.

(a) **(b)**

Figure 9 Comparison between the aerial photo and photorealistic 3D imagery of Trump International Hotel in Las Vegas (a) Aerial photo taken from an airplane (b) 3D reconstruction of the building

The first step in the process of generating 3D reconstruction of buildings is to stitch various images of similar scenes, which are indeed taken from different angles and scales.

b. Use the input images which are taken from the Camp Nou stadium in Barcelona, and find the stitched image containing all parts.

III. Street View

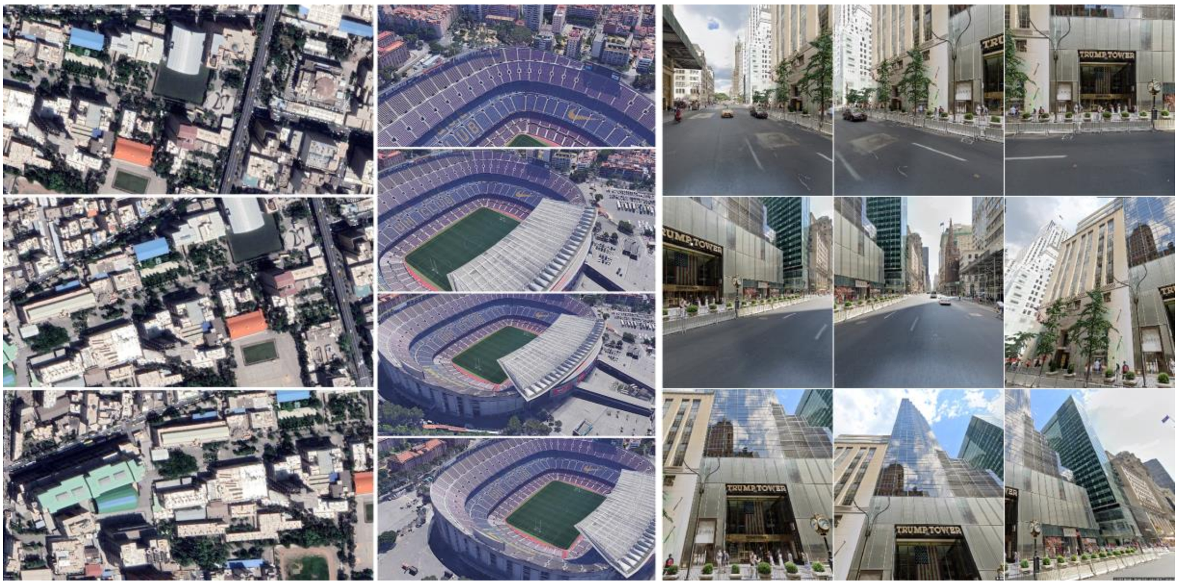
Another cool feature which was added to Google Earth in its 2008 release is the Street View. This feature enables user to view 360° panoramic streel-level photos of major cities and their surroundings. These photos are taken by cameras installed on automobiles, and can be viewed at different scales and from many angles.

(a) **(b)** **(c)**

Figure 10 Each Google car is equipped with cameras which simultaneously take images in multiple directions (a) A street view car (b) A close-up view of the rosette, which is made up of 15 cameras (c) A visualization of the spatial coverage of each camera, with overlapping regions shown in darker gray

While driving, Google cars capture multiple images in different directions at specific time intervals. More specifically, each car is equipped with a multi-camera rosette which is made up of 15 cameras (Figure 10). The images taken at each shot are later overlapped and stitched together into a single 360-degree image.

c. Several images taken in the vicinity of Trump Tower in New York City are given. Create a single panoramic image of the area.



(a) **(b)** **(c)**

Figure 11 All input images given for each of the three sections (a) Satellite images of Amirkabir University (b) Aerial photos taken from Camp Nou in Barcelona (c) Several shots taken from the entrance of Trump Tower in NYC

Note: Except for the final step (blending), you are allowed to make use of built-in functions and libraries. However, you may need to tune some parameters manually to obtain the best possible output.

4. Bringing Zombies Back to Life! (20 Pts.)

Keywords: *Image Enhancement, Image Restoration, Image Interpolation*

Taking photographs of a dead person may seem nasty and unpleasant to you. However, in Victorian England, it was quite common for families to take photos of their recently deceased loved ones to glorify their memories. In this tradition, known as *Post-mortem photography*, dead bodies were made to be in a natural gesture (sometimes even with their eyes open) with casual dress to make them look more lifelike.

Figure 12 In post-mortem photography, the photographer attempted to make the dead figure appear as natural as possible

Most of these photos were taken through *daguerreotype* process, in which a highly detailed image was created on a sheet of copper plated with a thin coat of silver, causing the surviving photos suffering from various image degradations. Moreover, after about two centuries, some of these photos have been subject to physical damages and degradations, including folding, tearing, handwritten text captions and various types of noises.

You are now expected to be able to handle almost all types of degradations in an image. In order to verify this, four post-mortem images are given (but hey, don't be scared!), each with <u>at least</u> two main degradations. Assume you were asked to enhance and restore them to a quality good enough to be published in a notable news website.

(a) (b) (c) (d)

Figure 13 Several post-mortem photos with different types of degradations (a) The standing girl in the middle is not alive. However, the photographer tried to make her look more alive by drawing on her eyes, so that they appear to be open. (b) The dead man was made to hold his daughter, one last time (c) Two sisters, one alive, one dead. The dead one (right) is being propped up with a special device (d) A Victorian couple poses with their recently-deceased daughter

Hint: Image interpolation can be used in order to restore torn parts, reduce the effect of folding and remove handwritten captions. Also various image enhancement and restoration techniques (e.g. histogram equalization, Gaussian filtering, etc.) may be come in handy to improve the visual quality of the images. However, additional techniques are required to remove some of the undesired effects (like stains on the photos).

Note 1: Your attempts must considerably improve the quality of the input images. In other words, applying various techniques blindly without a noticeable effect won't be of much worth.

Note 2: Include all intermediate results in your report.

Note 3: You are free to make use of all methods you've learnt in this course.

Note 4: All functions and libraries are allowed to be used.

5. Some Explanatory Questions (10 Pts.)

Please answer the following questions as clear as possible:

a. What is the difference between additive noise and multiplicative noise? Which one does "salt-and-pepper" noise belong to?
b. Imagine you want to remove a periodic noise from an image, and you are not allowed to use frequency domain filtering. What type of filter are you going to use? How does the frequency of the noise affect your kernel size?
c. Imagine you want to remove a Gaussian noise from an image, and you are not allowed to use spatial domain filtering. What type of filter are you going to use? How does the mean and variance of the noise affect your filter parameters?
d. Is image warping a reversible operation?
e. Write an appropriate transformation matrix for a transformation consisting of a vertically scaling of 1.5x, a counter-clockwise rotation by 45 degrees and a horizontally translation by 30 pixels.

Good Luck!
Ali Abbasi