Abstract

This document aims at giving a clear explanation of the different key parts of this project, so that anyone who wants to contribute to it may get up to speed in a minimal time. Furthermore, we define in this paper standards that need to be followed in order to allow easy collaboration.
## Contents

1. Introduction 3  
2. Motion estimation 4  
3. Image Reconstruction 5
1 Introduction

In many applications today, image resolution has become very important. While being a strong marketing argument for entry-level digital cameras, a high resolution sensor can greatly improve image quality if combined with good optics on a high-end digital SLR camera. What’s more, higher resolution in surveillance cameras can make it possible to see much more details, which could for instance help an investigation. Unfortunately, increasing the resolution at the sensor level implies buying new equipment, which might be financially prohibitive.

The goal of Super Resolution is to increase the resolution of an image by using many images very similar to it. Note that in this paper, a higher resolution image is not only an image that has more pixels, but is an image that has a greater resolution power (i.e., more details are visible). From these similar pictures of the same scene, where the camera has moved very slightly, we can gather all the information necessary that will in turn enable us to reconstruct a higher resolution image. The ideal scenario is if we can obtain four images, where the three last are shifted from the first one respectively by 0.5 pixel on the x-axis, by 0.5 pixel on the y-axis, and both 0.5 pixel on the x and y-axis. This makes it then possible to double the resolution in both dimensions without any kind of interpolation.

Super Resolution consists of two main steps: image registration and image reconstruction. The first estimates the motion between the different pictures, while the latter uses this information in order to reconstruct the high resolution image.
2 Motion estimation

Motion estimation techniques should only return the translation and rotation information, in the following format:

\[
[\delta_{est}, \phi_{est}] = \text{method}(\ldots)
\]

Where:

\[
\delta_{est} = \begin{bmatrix}
0 & 0 \\
3 & 10 \\
\ldots & \ldots \\
2 & 5 \\
\end{bmatrix}
\]

\[
\phi_{est} = \begin{bmatrix}
0 \\
25 \\
\ldots & \ldots \\
3 \\
\end{bmatrix}
\]

All outputs must be formatted exactly in this way in the implementation itself of the MATLAB function.

Note: The MATLAB function \texttt{imrotate} and \texttt{circshift} work with the same conventions, i.e.:
imrotate(im, 25, 'crop') and circshift(im, [3, 10])

**However**, the custom `shift` function included in this project takes the parameters in the opposite way, i.e.:

`shift(im, 10, 3)` in order to achieve the same result.

### 3 Image Reconstruction

Reconstruction techniques should accept the input images as 2D “grayscale” images\(^1\), packed in a cell data structure, given by:

\[
im = \{im_1, im_2, im_3, im_4, \ldots\}
\]

Images can then be extracted from the cell using the MATLAB command: `im{i}`, where \(i\) is the image number.

---

\(^1\)This means that an RGB image must first be divided into three 2D matrices and the function will reconstruct each “layer” individually.