

# Compression and File Sharing of Facial Images via Laplacian Networks

## 1 Background

Applications such as Snapchat and Facebook provide streamlined platforms for sending facial images between users. However, due to the high volume of images transferred between users on these platforms, images sent through these applications are either downsampled or captured at a lower resolution than the resolution of a user's camera. This facilitates lower transfer times for the sender, but results in inferior-quality photos for the recipient.

## 2 Aim

To address the limitations presented above, the use of a Laplacian Pyramid Super Resolution Network (LapSRN) is proposed to quickly upsample a downsampled or natively low-resolution facial image for a receiving user. Allowing images to be transferred at an extremely low resolution will reduce the cost of data transfer and improve latency for users.

## 3 Methodology

The methods that will be utilized to complete this project are outlined in this section.

### 3.1 Components

The components for this project, including libraries, hardware, and the LapSRN network are listed below. All code will be implemented in Python.

1. A LapSRN coded in the Pytorch deep learning library
2. A graphical user interface coded in tkinter and opencv

### 3.2 Methods

The first step of this project is to create a working LapSRN in Pytorch. The discriminators in the LapSRN will use the Leaky ReLU activation function as they mitigate the effects of ReLUs dying when a large gradient is passed through. A pretrained network will be further trained on facial images. The training data for the network will be the CelebFaces Attributes Dataset (CelebA). The images will be split into training and testing sets, downsampled, centered, and fed into the network. The output will then be compared to the ground truth images. The image upscaling process is illustrated in Figure 1.

After the scripts for interfacing with the LapSRN are complete, an application will be created that facilitates communication between machines. The application will utilize opencv and tkinter to interface with the webcam and create the user interface respectively.

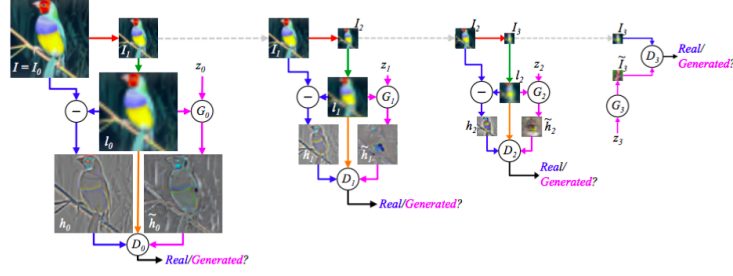


Figure 1: The image upscaling process carried out by the LapSRN. An input is fed into the right-most level of the pyramid. The process of extracting nonlinear features and then upsampling via transposed convolutional layers is repeated at each level of the pyramid. Though this diagram shows three such layers, the adaptation of the LapSRN implemented in this project only has two levels, yielding an upscaled image 4x upscaled vs 16x upscaled in a network with three layers.

## 4 Term Project Summary

During the weeks leading up to the end of the semester, there are four aspects of the project that were started and then completed: creating a LapSRN in pytorch, creating scripts for data preprocessing, writing the software so that the application can save and load images upscaled in different manners to facilitate comparison, and writing the software for the user interface.