Vision`

Mohannad Atmeh
Abdel-aziz Hamdan
Mohammad Ibrahim Abu-Amara

Introduction

Imagine taking an amazing picture, and you get home exited to put that picture on a huge monitor in your living room, so you can show it off to people who come to your house, to find out that when you put a small picture on a big screen, most of the details in the picture are lost, which leads to a blurry picture, that does not look good. Now that happens when you just have fun and take normal pictures, imagine when that picture is important to a crime, or for getting important information, or a memory of someone, you can't just get back in time to get a better looking picture with higher resolution, the time machine has not been invented yet, and we also can't just try to guess what that blurry part of the picture is and assume the information we need. We have to solve this problem, which is not limited to only pictures, but videos as well.

That problem is called the Super Resolution problem, SR for short, which is essentially just taking a low resolution picture and converting it to a high resolution picture, with magic of course. It might seem like an easy problem to solve; because of the fact that the requirement are simple, but it's not easy, it's simple, but not easy, there are a lot of details that we have to care of in order to produce the same details in both the lower resolution picture and the higher resolution one, to give an example, if the lower resolution picture is a picture of a licence plate of a car, we have to make sure that we keep that license plate correct and not lose some information from it.

System Description

Our system is quite simple to describe, which also leads to better scalability, higher availability and fault-tolerance, and an efficient way to solve the problem. The system has three main components,

1. User Interfaces
2. Distributed System of processing units
3. Deep learning Models

The user interface, also called the front-end, is responsible to take care of the full user experience, from uploading the picture or the video, to displaying the new higher resolution image or video. It will be a Flutter application, which handles the mobile version, both IOS and Android, Desktop Application for Windows, macOS, and Linux, as well as a web version. This will make sure that we cover almost all platforms, so people can use it whenever they want, fast and easy.

The distributed system of processing units will handle all the temporary data operations, why temporary?, might ask, because these units will be stateless meaning that they shouldn't know which request is which and user data or save anything. These processing units should also make sure the whole system is resilient, in case of a failure, as well as handling huge amounts of requests with a small amount of resources.

The deep learning Model, the cherry on the cake, will have our main algorithms and learning models to convert the user's lower resolution picture or video to a high resolution picture or video, using state-of-the-art deep learning models, and APIs. Why are we using deep Learning, not just normal image processing algorithms ? or why not use the traditional methods to enhance resolution ?, you might ask, well first of all, traditional methods don't really work for the general case, meaning that they are restricted to certain types of pictures, and even with that, they don't always keep the details of the picture that we talked about earlier. Second of all, deep learning models, or even normal Machine learning models, learn from every picture that are fed into them, that is why they are called learning models, which leads to automatic development and scale, and make it adaptable to new types of data. In addition, the problem itself isn't traditional, we can't just create an algorithm that always works fast, and achieve a high resolution picture with all the important details, we don't know how to fill the extra pixels in the image after we convert it to a high resolution, so we need some sort

of learning model to choose the best thing to fill those pixel, and also keep learning on it’s own, based on previous positive experience.

System Purpose

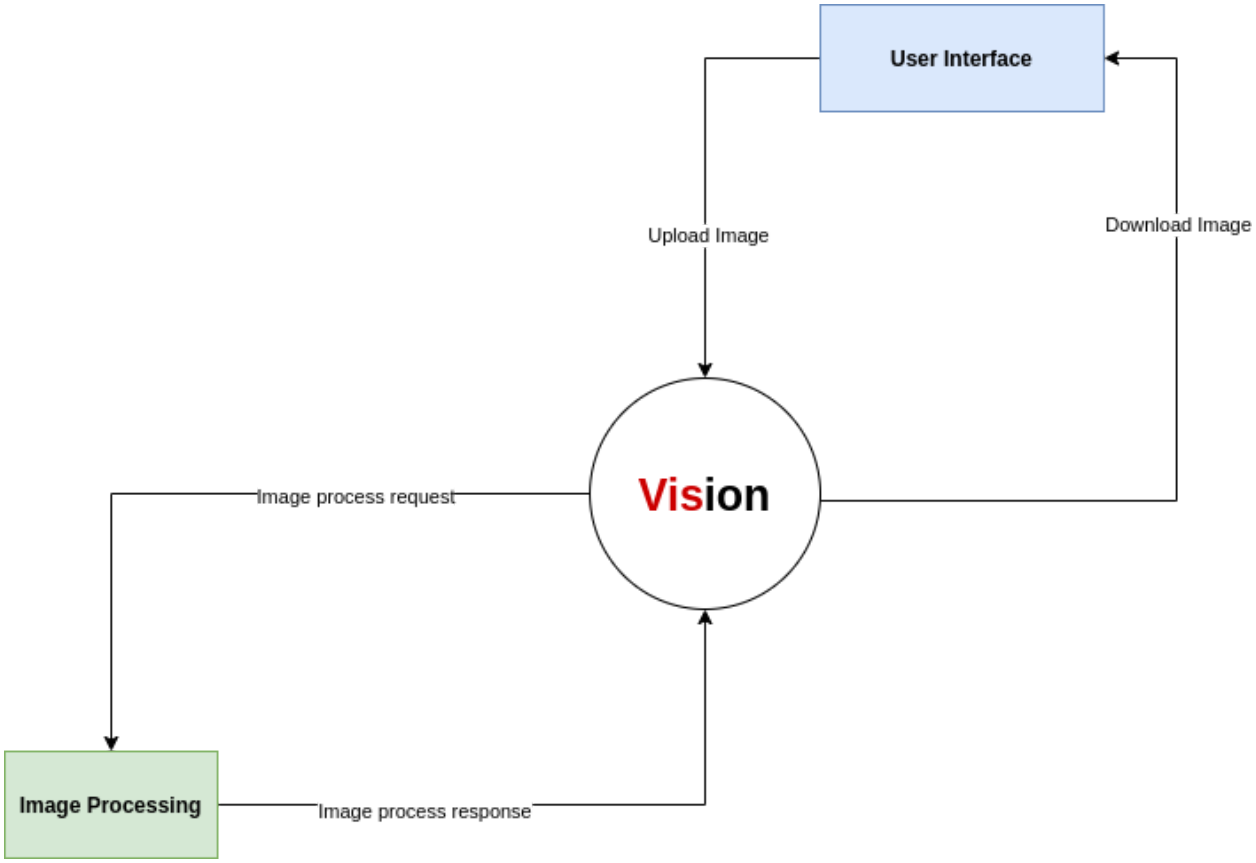
The purpose of the system is simple, the system takes an image or a video from the user, processes it, and finally returns the processed image or video to the user. This, of course, needs to happen in milliseconds, even when the system is under load. And when the system experiences failure, it needs to meet it with elegance rather than disaster, and stay highly responsive, giving users effective interactive feedback. This is because our system has critical values to the users, and they need the results as soon as possible. Also one thing that we should provide is security, we don't want other people getting into users data, so we are going to provide a stateless application to make sure we, the system, don't have data to leak or something like that. As well as, making sure that our network interactions with the user is also secure, to be able to receive and send data securely.

Problem Statement

The main problem is converting a lower resolution (lower Pixels per inch) image, to a higher resolution (high Pixels per inch) image. Which also can be extended to videos, because of the fact that a video is just a sequence of images called frames. And it's the same for our requirement, Pictures and DVR videos.

So if we have a frame, image or a frame from a video, called X, we have to return X' ; such that X' have a higher resolution than X with the details from Xexisting in X'. So given a frame, produce a frame of larger size, and with noticeably more pixels, as well as higher image quality.

The System Context View



Literature Review

Deep Learning for Single Image Super-Resolution: A Brief Review

This review brings a new perspective on how we should look at the deep learning models on solving problems, but in the context of the Super Resolution problem of course. It reviews representative deep learning models that relate directly to the Single Image Super Resolution, SISR for short, and then group them into two colors, relative to their contribution to two core aspects of SISR, the first one being, the exploration of efficient neural network architectures for SISR, and the second being, the development of effective optimization objectives for deep SISR learning. It highlighted some benchmarks of deep Architecture for SISR, which show some amazing numbers of how deep learning enhances the SISR operation.

Image Super-Resolution Using Deep Convolutional Networks

This paper proposed a deep learning method for single image Super Resolution, it directly learns an end to end mapping from the low resolution image. That mapping is done using Convolutional Neural Networks, CNNs for short. It showed an amazing proof of how traditional sparse-coding based Super Resolution methods can also be seen as CNNs, however, unlike traditional methods that handle each component on it's own, their deep learning model jointly optimizez all layers and components, with a lightweight structure, and maintaining state-of-the-art quality. All of that with the idea of working efficiently and having real online applications across the industry.

Photo-Realistic Single Image Super-Resolution Using GANs (Twitter)

Twitter researchers, Twitter Cortex research group to be exact, showed that some deep learning methods for Super Resolution are good, but there are some issues in them, despite the breakthrough in accuracy and speed that people got from Convolution Neural Networks. One of the questions that remains unanswered is, how do we recover the finer texture details when we super resolve at large upscaling factors? So they introduced SRGAN, a Generative Adversarial Network for Super Resolution that solves this problem.

iSeeBetter

Three researches figured out that although the learning-models have enhanced the performance of a single picture Super Resolution, yet applying those methods, the single picture Super Resolution methods, consecutively to each video frame is going to cause a lack of temporal coherency, which is something that we don't want to have for sure. Convolutional Neural Networks, CNNs, outperformed traditional methods in almost every way, but Generative Adversarial Networks, GANs, offers a competitive point for us, which is begin able to mitigate the issue of a lack of finer texture details, usually seen with CNNs when super resolving at large upscaling factors. So iSeeBetter used GANs in order to be able to produce Super Resolution videos, VSR, that renders temporally consistent VSR.

RAISR: Rapid and Accurate Image Super Resolution (Google)

This is my personal favorite paper about Super Resolution, because it really shows how many important applications the problem has, not just the fictional movies image enhancement kind of thing, to give a few about the applications of the Super Resolution, Google showed it in many of its products, the Pixel 2 phone used it in its digital zoom, which showed amazing result over the years. Google+ (RIP) used up to 75% **less** bandwidth per image when using RAISR, and they're applying RAISR to more than 1 billion images per week, reducing these users' total bandwidth by about a third. Not just that, it also produces high quality restoration while still being faster than the current leading algorithms by a factor of two, with extreme low memory footprint. This way it can be executed on a smartphone offline, rather than on a server or a heavy machine to handle it.

Challenges

1. ### Brand New!

   Students don't normally just get into AI and Machine learning, and whatever is around it, easily. So it was hard to understand these kinds of stuff, and how to be able to connect the basics that we know, from math basics, computer basics, with what we are trying to solve. We can easily find working code online, but we also would like to understand the Math behind it, the Theory, its Applications, its History, and all that sweet stuff that makes Computer Science what it is.

2. ### Simple yet Hard to solve!!

   The way I think about it is that it's trivial to explain to anyone what the Super Resolution is, like for real, get me a group of 3 year olds and I am sure that 99% of them will understand it. But algorithmically, it is very hard to solve, yet we need an algorithm that works on a smartphone or something to that extent, with a small amount of memory, which is, again, "sweet stuff that makes Computer Science what it is", so we are glad to be able to try and solve these kinds of problems.

3. ### Proof of Concept, PoC, without a working application is hard!!!

   This is a combination of Simple yet hard and Brand New, like yeah we know that this kind of deep working algorithms are used in high tech companies and it works, but how to prove that it will work in our case, or just even prove it at all, so we need to implement it and show that it works in most cases at least; because the use of it will be important to so many people, especially for the requirement of our system.

Projection

NOTICE THAT THIS IS ONLY FOR PHASE ONE

Gantt Chart

Vision Gantt Chart

Gantt Chart

WBS NUMBER	TASK TITLE	START DATE	DUE DATE	DURATION	PHASE ONE
					25 26 27 28 1 2 3 4 5 6 7 8 9 10 11
1	**Project Research**				
1.1.0	Main Problem	25-Feb-2021	28-Feb-2021	4	
1.1.1	Research Papers	27-Feb-2021	02-Mar-2021	4	
1.1.2	Real life projects	28-Feb-2021	04-Mar-2021	5	
1.2.0	Introduction	02-Mar-2021	03-Mar-2021	2	
1.2.1	System Description	02-Mar-2021	03-Mar-2021	2	
1.2.2	System Purpose	03-Mar-2021	04-Mar-2021	2	
1.2.3	Problem Statement	03-Mar-2021	04-Mar-2021	2	
1.2.4	The System Context View	05-Mar-2021	05-Mar-2021	1	
1.2.5	Literature Review	05-Mar-2021	08-Mar-2021	4	
1.2.6	Challenges	07-Mar-2021	08-Mar-2021	2	
1.2.7	Projection	08-Mar-2021	09-Mar-2021	2	
1.2.9	Verification	09-Mar-2021	11-Mar-2021	3	

Vision

Network Diagram

Vision Network Diagram