Spring 2022

Super-High Resolution Imaging Using Easy Accessible Resources

Jamison Murphy
Loyola Marymount University, jamisonmurphy24@gmail.com

Follow this and additional works at: https://digitalcommons.lmu.edu/etd

Part of the Computer Sciences Commons

Recommended Citation
https://digitalcommons.lmu.edu/etd/1149

This Research Projects is brought to you for free and open access by Digital Commons @ Loyola Marymount University and Loyola Law School. It has been accepted for inclusion in LMU/LLS Theses and Dissertations by an authorized administrator of Digital Commons@Loyola Marymount University and Loyola Law School. For more information, please contact digitalcommons@lmu.edu.
Super-High Resolution Imaging Using Easy Accessible Resources

Jamison Murphy (MS Comp. Sci.)
SYEG 696 Capstone Presentation
Acknowledgements

- Dr. B.J. Johnson, PhD, CMSI Graduate Program Director
- Dr. Ray Toal, PhD, Capstone Advisor
- Dr. Mandy Korpusik, PhD, Technical Advisor
- Dr. Lei Huang, PhD, Technical Advisor
Agenda

- Summary
- Background
- Motivation
- Problem
- Goal
- Methodology
- Measurement of Effectiveness
- Verification Requirements
- Stakeholders
- Scope
- Solutions

- Solution Architectures
- Demonstration
- Solution Analysis
- Recommended Solution
- Recommendations
- Scheduling
- Conclusion/Lessons Learned
- Future Opportunities
- Accomplished
- Listed Sources
Background: Satellites, drones, and other aircrafts are used to present aerial coverage views of what’s occurring in communities by taking images of them that can be seen on software using graphics.

Problem: Many generic systems and hardware for non-governmental and low budget users are not capable of executing software producing super-high resolution images from these aerial and space crafts because of their lack of capability to process large amounts of data.

Goal: Produce super high resolution images on systems and hardware that are not using expensive and exclusive to governmental officials.

Methodology: Agile software development strategies will be implemented to test and analyze solutions for image resolutions.

Opportunity: Images will be a lot clearer and the processing of them will be simplified with low cost as production companies look towards less expensive systems.

Accomplished: Super high resolution images were produced at fast speeds, good image clarity, and less complexity using cloud platforms.

Conclusion: Image resolutions with super high quality can be obtained on cheap systems and hardware.
Researchers and photographers use satellites, drones, and other aerial crafts to take images of communities and environments for studies and marketing:

- detecting environmental changes over time
- showcasing the communities for 3D maps
- provides clear observation of a scene during 911 emergency
GPU’s (Graphical Processing Units) are hardware that allows computers to process graphics for images, videos, and videogames.

The first official GPU was the GeForce 256 created by NVIDIA.

A typical GPU is expected to process a “minimum of 10 million polygons per second” [1].

Over recent years of development, they have become “the largest, most complex, and most expensive components” for any technological device [3].

Since the “flurry of new graphics competitors” popped up in 1993, that number has slimmed down to 3 main producers [2]:

- NVIDIA: $495 – 1,823
- AMD: $259 – 999
- Intel: $???
Cloud platforms are services that provide servers for working on and storing files to prevent having a significant amount of CPU and RAM usage.

Google Colab is a cloud platform that allows users to upload, create, edit, execute, and store their Python files in Jupyter Notebook format and providing fast online GPU’s for large data projects.

Replit is a cloud platform that allows users to upload, create, edit, execute, and store files of any language up to date and provide a GPU.

GPU’s available in Colab:
- Tesla T4
- Tesla K80
- Tesla P4
- Tesla P100

Colab Cost: Free - $49.99
Replit Cost: Free
Motivation

- Interest in image processing
- Urge to find ways to do computing tasks a lot more simplified and cheaper
Many generic systems and hardware for non-governmental and low budget users are not capable of executing software producing super-high resolution images from these aerial and space crafts because of their lack of capability to process large amounts of data.

- Blocked, suspended, or unrecognized processes on CPU’s
- Reduction of image sizes from high dimensions to low dimensions
Goal

• Produce super high resolution images on systems and hardware that are not expensive and exclusive to governmental officials
Methodology

- Agile software development – the utilization of multiple tests, executions, or “sprints to create and deliver a large software project
- Kanban – an agile software development strategy where multiple tests are being executed on a visual schedule leading up to a delivery
Measurement of Effectiveness (MOE)

- **Image Clarity** - How clear are the images coming out
- **Processing Time** - How long does it take for the images to be fully processed and displayed
- **Complexity** - How difficult it is to execute
- **Cloud Compatibility** - The ability to run on a cloud platform service
Verification Requirements

- Image Clarity: Blurriness is reduced and pixel quality enhanced
- Process Time: <10 min of processing output
- Complexity: In between moderate and easy
- Cloud Compatibility: Can execute on Colab or Replit
Verification Requirements

Image Clarity
Verification Requirements

Processing Time

- 59s completed at 8:53 AM
Verification Requirements

Complexity

*This is a list of cells or steps that need to be executed in order for the programming process to be completed.
Verification Requirements

Cloud Compatibility

*Shows packages fully installed on platform*
Stakeholders

• Non-Governmental Users
  ~ Drone Users
  ~ Undergraduate Earth Data Scientists/Researchers
  ~ Photographers
  ~ Landscapers
• Companies
  ~ Google
  ~ Anaconda
  ~ Replit
• Client
  ~ LMU Marketing & Communications
## Scope

<table>
<thead>
<tr>
<th>In-Scope</th>
<th>Out-of-Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Improvement of image resolution from aerial &amp; space coverage</td>
<td>• Photographers using improved software to improve image</td>
</tr>
<tr>
<td>• Increased capability of software to process super resolutions</td>
<td>resolution of ground-level images</td>
</tr>
<tr>
<td>• Reduced expenses for advanced hardware to detect images</td>
<td>• Biologists using improved software for volume-metric</td>
</tr>
<tr>
<td></td>
<td>imaging</td>
</tr>
<tr>
<td></td>
<td>• More clarity to 3-D imaging</td>
</tr>
</tbody>
</table>
Neural networks are numerous algorithms grouped into layers that are used to detect characteristics in large datasets such as image and video pixels, audio, and physical motion.

Testing a list of software packages with Python and C++ that are neural network compatible:

- Python (w/ PyTorch)
- Python (w/ Tensorflow)
- Python (w/ Cognitive Toolkit (CNTK))
- Python (w/ Gluon)
- C++ (w/ Caffe)
Solution #1 - Python (w/ PyTorch)

• PyTorch is a machine learning package created by the Facebook AI Research team that utilizes torch libraries in Python for handling tasks in computer vision and natural language processing
TensorFlow is a machine learning and artificial intelligence package created by the Google Brain team that can be used in Python and C++ for a numerous amount of tasks but it specializes in deep neural networks.
• Cognitive Toolkit (CNTK) is a deep learning package for Python and C++ created by the Microsoft Research team that presents neural networks in computational steps which are illustrated through a directed graph.
Gluon is a package created by Apache that is driven by models and helps deliver the best API services for quick network processing.

Solution #4 - Python (w/ Gluon)
Solution #5 - C++ (w/ Caffe)

- Caffe is a package created at the University of California, Berkely that specializes in deep learning networks using C++
Example of a Super Resolution Convolution Neural Network (SRCNN)

Solution Architectures – Operational View

Figure 2: Sketch of the SRCNN architecture.
Solution Architectures – System View

Input Images ➔ Uploaded ➔ Colabrapory ➔ Send Images ➔ TF ➔ Send Images ➔ GLUON ➔ Send Images ➔ Export HR Images

Input Images ➔ Uploaded ➔ Replit ➔ Send Images ➔ Caffe ➔ Send Images ➔ Export HR Images
Demonstrations
## Solutions Analysis

<table>
<thead>
<tr>
<th>Suggested Languages</th>
<th>Image Clarity</th>
<th>Processing Time</th>
<th>Complexity</th>
<th>Cloud Compatibility</th>
</tr>
</thead>
<tbody>
<tr>
<td>Python ( w/ PyTorch)</td>
<td>Good</td>
<td>0.26 min/att</td>
<td>Easy</td>
<td>Yes</td>
</tr>
<tr>
<td>Python (w/ Tensorflow)</td>
<td>Ok</td>
<td>0.47 min/att</td>
<td>Moderate</td>
<td>Yes</td>
</tr>
<tr>
<td>Python (w/ CNTK)</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>No</td>
</tr>
<tr>
<td>Python (w/ Gluon)</td>
<td>N/A</td>
<td>N/A</td>
<td>Hard</td>
<td>Yes</td>
</tr>
<tr>
<td>C++ (w/ Caffe)</td>
<td>N/A</td>
<td>N/A</td>
<td>Hard</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Ranges
- **Image Clarity:** Poor < Ok < Decent < Good < Great
- **Complexity:** Hard < Moderate < Easy
Python (w/ PyTorch) is the best choice for super high resolution images because it has good image quality, the fastest speed, easy to use, and cloud compatibility.
Recommendations

• When running PyTorch, try to keep dimensions of lower resolution images between 128 x 128 and 256 x 256
• To get the fastest processing speed on Colab, check to see if your GPU is Tesla P100
• Use image format .jpg or .png for input
# Scheduling

**Super-High Resolution Imaging Using DeepSUM**

**Project Details**
- **Project Start:** Tue, 1/11/2022
- **Today:** Thu, 4/28/2022
- **Display Week:** 8

## Tasks

<table>
<thead>
<tr>
<th>Task Description</th>
<th>Assigned</th>
<th>Progress</th>
<th>Start</th>
<th>End</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design &amp; Execution: Install and test all neural network packages/modules from libraries w/sample data</td>
<td>100%</td>
<td>2/13/22</td>
<td>2/22/22</td>
<td></td>
</tr>
<tr>
<td>Test solutions #1 &amp; 2 w/ client-given data, evaluate and analyze results</td>
<td>200%</td>
<td>2/23/22</td>
<td>3/1/22</td>
<td></td>
</tr>
<tr>
<td>Test solution #3 w/ client-given data, evaluate and analyze results</td>
<td>100%</td>
<td>3/2/22</td>
<td>3/8/22</td>
<td></td>
</tr>
<tr>
<td>Test solution #4 w/ client-given data, everyone and analyze results</td>
<td>100%</td>
<td>3/9/22</td>
<td>3/23/22</td>
<td></td>
</tr>
<tr>
<td>Test solution #5 w/ client-given data, everyone and analyze results</td>
<td>100%</td>
<td>3/16/22</td>
<td>3/22/22</td>
<td></td>
</tr>
<tr>
<td>Implementation: Decision made on determining the best solution</td>
<td>100%</td>
<td>3/23/22</td>
<td>3/29/22</td>
<td></td>
</tr>
<tr>
<td>Attempt at world record</td>
<td>0%</td>
<td>3/30/22</td>
<td>4/5/22</td>
<td></td>
</tr>
</tbody>
</table>

![Gantt Chart Image](image.png)
Conclusions/Lessons Learned

- Image resolutions with super high quality can be obtained on cheap systems and hardware.
  - Simplicity and cloud compatibility is the best going forward with reduced instruction set computers (RISC) such as PC’s.
  - What is highly promoted in software packages might not be the most efficient and in a good quality.
Future Opportunities

• Images will be a lot clearer and the processing of them will be simplified with low cost as production companies look towards less expensive systems

~Less expensive equipment to detect laser-scanned images in the biological and medical fields and get very good results
~Less expensive equipment in photo editing for earth data scientists and photographers
Super high resolution images were produced at fast speeds, good image clarity, and less complexity using cloud platforms.
Listed Sources


