CONTINUOUS AUTOMATIC DETECTION OF CRACKS IN TUNNELS USING MACHINE MEARNING AND ARTIFICIAL INTELLIGENCE TECHNIQUES FOR SAFETY MONITORING

FINAL PROJECT REPORT

by

Fred Daneshgaran and Marina Mondin

California State University Los Angeles

Sponsorship

UTC-UTI

For

University Transportation Center for

Underground Transportation Infrastructure

(UTC-UTI)

July, 2020

Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated in the interest of information exchange. The report is funded, partially or entirely, by a grant from the U.S. Department of Transportation's University Transportation Centers Program. However, the U.S. Government assumes no liability for the contents or use thereof.

<table>
<tr><td>1. Report No.</td><td>2. Government Accession No.</td><td colspan="2">3. Recipient's Catalog No.</td></tr>
<tr><td colspan="2" rowspan="2">4. Title and Subtitle
Continuous Automatic Detection of Cracks in Tunnels Using Machine Learning and Artificial Intelligence Techniques for Safety Monitoring</td><td colspan="2">5. Report Date: July 2020</td></tr>
<tr><td colspan="2">6. Performing Organization Code</td></tr>
<tr><td colspan="2">7. Author(s)
Fred Daneshgaran (0000-0003-0120-143X) and Marina Mondin (0000-0003-3105-3802)</td><td colspan="2">8. Performing Organization Report No.</td></tr>
<tr><td colspan="2" rowspan="2">9. Performing Organization Name and Address
University Transportation Center for Underground Transportation Infrastructure (UTC-UTI. Tier 1 University Transportation Center. Colorado School of Mines. Coolbaugh 308, 1012 14th St., Golden, CO 80401</td><td colspan="2">10. Work Unit No. (TRAIS)</td></tr>
<tr><td colspan="2">11. Contract or Grant No.</td></tr>
<tr><td colspan="2" rowspan="2">12. Sponsoring Agency Name and Address
United States of America. Department of Transportation. Research and Innovative Technology Administration</td><td colspan="2">13. Type of Report and Period Covered</td></tr>
<tr><td colspan="2">14. Sponsoring Agency Code</td></tr>
<tr><td colspan="4">15. Supplementary Notes
Report also available at: https://zenodo.org/communities/utc-uti</td></tr>
<tr><td colspan="4">16. Abstract
Structural monitoring of the inner walls of a tunnel, both during construction and use, is important for safety, performance, liability, and cost. The goal of this project is to develop a fully automated system to perform continuous monitoring of tunnels during and after construction. The following intertwined aspects to this problem require further study: investigation and development of (a) a suitable technique for relevant data acquisition (this usually involves imaging the interior walls of the tunnel, but the exact method for doing this that is reliable, cost efficient, and easily automated is an open problem); (b) tools and techniques to acquire the data in an efficient and automated manner and communicate the information to proper processing center; (c) suitable techniques for post processing of acquired data to generate an intermediate observation space on which detection algorithms may be applied; (d) suitable algorithms and techniques for pattern detection and classification; (5) the software suite needed for data processing from post processed data to implement selected pattern detection and classification algorithms/techniques, assess their performance and develop recommendations.</td></tr>
<tr><td colspan="2">17. Key Words
Data Acquisition, Pattern Recognition and Machine Learning, Deep Convolutional Neural Networks</td><td colspan="2">18. Distribution Statement
No restrictions.</td></tr>
<tr><td>19. Security Classification (of this report)
Unclassified</td><td>20. Security Classification (of this page)
Unclassified</td><td>21. No of Pages
44</td><td>22. Price
NA</td></tr>
</table>

Table of Contents

List of Figures

List of Tables

List of Abbreviations

UTI: Underground Transportation Infrastructure
CNN: Convolutional Neural Network
GPS: Global Positioning System
IPT: Image Perspective Transformation
CCD: Charge Coupled Device
DSLR: Digital Single Lens Reflex
NN: Neural Network
GPU: Graphic Processing Unit
LiDAR: Light Detection and Ranging
NDE: Non Destructive Examination
AI: Artificial Intelligence
SLAM: Simultaneous Localization and Mapping
RADAR: Radio Detecti0on and Ranging
ToF: Time of Flight
vSLAM: Visual SLAM
LED: Light Emitting Diode
LCD: Liquid Crystal Display
SFM: Structure From Motion
DFC: Depth Field Combining
RGB: Red Green Blue
PnP: Point and Perspective
CUDA: Compute Unified Device Architecture
FPS: Frames Per Second
ML: Machine Learnning

CHAPTER 1 – INTRODUCTION

1.1. Overview

The increased urbanization and population density in major urban centers of the world has led to a greater demand for new underground transportation infrastructures. Indeed, a well-organized network of tunnels in a densely built-up urban area is able to solve problems such as traffic congestion, noise and air pollution. Nevertheless, design, construction and maintenance of underground infrastructure still rely on empirical methodologies that most of the time result in a cost increase and extended handling times. In this context, where automatic assessments could be an important aid for maintenance of tunnels, the R&D conducted focuses on an affordable and reliable low cost, crack detection system as replacement or augmentation to traditional visual inspection techniques.

Underground Transportation Infrastructure (UTI) are characterized by poor lighting conditions and dirty surfaces covered by dust and mold. Moreover, most of the time huge scuffs on the walls, due to the aging of the lining, could easily resemble the shape of a crack. Due to these main reasons, automated inspection of tunnels is still an open challenge with not so many research studies and few enterprises that offer automated solutions.

UTI inspection policies and procedures lack a nation-wide standard concerning reliability and frequency of maintenance. The most commonly adopted methods are visual inspection and hammer sounding, providing only qualitative information and requiring the presence of multiple skilled operators. Moreover, such procedures usually need to stop traffic flow to be performed [1]. There is, therefore, a great interest in autonomous and automatic inspection systems, that would reduce the need for human operators, would allow performing inspections without blocking the traffic flow, would allow inspecting dangerous or toxic environments, and would, in general, reduce the maintenance costs. Possible fields of application of such systems could include maintenance of car, metro and train tunnels, of a possible future hyperloop, mines, chemical or nuclear plants, and sewer pipes to name a few.

In this multifaceted research, we aim to address the problem from ground up via design, development and implementation of data acquisition systems for monitoring of the interior lining of tunnels and other underground transportation infrastructure, training and using deep Convolutional Neural Networks (CNNs) for automatic classification of the acquired data for anomaly detection, and design and development of an autonomous indoor navigation system for data annotation in absence of any GPS or beacon signals of any kind.

1.2. Image and Video Based Data Acquisition System

Visual inspection of underground infrastructure is by far the most used technique for crack detection and tunnels maintenance. Several studies and solutions have been presented to automate and simplify this process but currently, these systems are known to be expensive and in many cases not reliable.

Cracks detection using image processing

Before 2010, Japan and South Korea were at the leading edge of research for crack detection in tunnels. Researchers in Japan proposed an automatic monitoring system to detect cracks in tunnels based on a mobile robot [2]. They used edge enhancement and graph searching technology to extract the cracks on images.

More recently, due to the high expenses which implies visual inspections, engineers were driven to develop more complete and reliable solutions. This is true especially for all those countries with a widespread network of underground infrastructures. That is the case of China where for example the network of subways in Beijing has been rapidly developed and should be maintained.

At present, urban rail transportation in China is still mainly dominated by manual checking, missing in efficiency and security. In this context, the *Beijing Municipal Commission of Education Beijing Jiaotong University* developed an algorithm to detect cracks in tunnels based on image processing [3].

The acquisition system is based on Charge Coupled Device (CCD) cameras with high-frequency data collection. A Laser is used as light source and a custom Image Perspective Transformation (IPT) is used as core of the module. *Toshiba Research Europe* in 2014 tried to investigate a low cost system using a technique known as Structure from Motion (SFM) in order to recover a 3D version of the tunnel surface and find defects with comparisons [4].

A different research project has been developed by *Pavemetrics Systems Inc.* making use of top grade and expensive equipment to acquire both 2D images, and high resolution 3D profiles of infrastructure surfaces at speeds of up to 30 Km/h (tests have been carried out at 20 Km/h) [5].

More recently, in 2016 a research has been carried out in order to automatically collect and organize images taken from tunnels carrying high-voltage cables [6]. The project is developed by making use of consumer Digital Single Lens Reflex (DSLR) cameras and high-power polarized LEDs, in order to improve image quality, mounted on a lightweight aluminium frame, which is studied to dampen vibrations during data capture.

Crack detection using Machine Learning

As a first attempt in 2002, during the golden age of Support Vector Machines, four researches proposed an efficient tunnel crack detection and recognition method [7]. No further recorded attempts have been made using machine learning algorithms until 2016, when for the first time a deep learning model has been used within the detection algorithm [8]. The system exploits the algorithm to devise an automatic robotic inspector for tunnel assessment. The robotic platform consists of an autonomous mobile vehicle with all sensors attached to a crane arm. Two sets of stereo-cameras are used for taking the necessary images and an Arduino Uno board is used as a pulse generator synchronizing the two cameras. The first stereo pair is responsible for the crack and the other defects detection that lay on the tunnel lining. The second stereo pair is used for the full 3D reconstruction of high fidelity models of the areas of cracks. A FARO 3D laser scanner is deployed when a crack is detected for a precise calculation of any tunnel deformation that could be present.

1.3. High Resolution Camera Based Data Acquisition System

Different solutions have been proposed during the past few years, even if this field as a whole is still at its infancy. The first proposed techniques in literature were based on image processing techniques, and not on deep learning algorithms, but can be considered very inspiring from the hardware point of view and for providing examples of different data acquisition systems. For example, in 2007 Hanyang University [9] built a robotic system able to collect images inside a tunnel and then process them in order to find cracks in the concrete. This mobile robot was remotely controlled in order to maintain constant distance from the tunnel walls and it was equipped with CCD cameras, in order to get images under the best possible conditions. The image capture system was also equipped with illuminators and encoders for computing speed and position of the robot in the tunnel. From the software point of view, the images were processed with computer vision techniques in order to find images with cracks and measure their characteristics. Even though the measurements of cracks were very good once detected, the system was very inefficient in crack detection, with an error rate of around 80 %. In another development, a Taiwanese team proposed a new method (*Image-mosaic technology*) for reconstructing 3D tunnel surfaces from images taken with cameras [10]. While the aim of the R&D conducted by the Taiwanese team was different from ours, the common element was the necessity of having a good database of images. In [4] the authors proposed the use of DSLR cameras for image acquisition; in particular, they built a circular array of five DSLR cameras arranged on a remotely controllable rail, in order to image all the internal surfaces of the tunnel. While the approach proposed is interesting and one of our prototypes also utilizes a DSLR camera, our aim is to develop a simple system at much lower cost point that is portable and can be mounted on any vehicle. An interesting work was presented in 2013 [11], when researchers created an acquisition system on a truck moving in normal traffic flow inside a tunnel in order to develop a system that does not require a complete blockage of the tunnel segment under scrutiny.

One of the first application of Convolutional Neural Networks (CNNs) to the problem of crack detection, was presented by a team in [12] which proposed a vision based method using deep neural architecture for classification of images with or without cracks. The proposed architecture however was not based on recent developments in deep networks and was composed of only 2-convolutional layers, with a last soft-max layer for the final classification of the pictures. In that work, the authors took more than 100000 images and used 80 % of them for the training set. The objective of the project was to compare the Neural Network (NN) based approach with other computer vision based algorithms and, by looking at the results, the authors asserted that NN technique is significantly better than many of the other image processing approaches.

One of the best results with these new techniques was achieved by Cha and Choi in 2017 [13]. The images for the training of the network had been taken with a DSLR camera inside a tunnel in Manitoba and more than 40000 images had been collected. In particular, 80% of the acquired images were used as training set, while 10% each were used for testing and validation. The results obtained by running the algorithm on two Graphic Processing Units (GPUs) were excellent, with around 97% accuracy in crack detection in the testing phase. Another work presented in the literature regarding the tunnel inspection is the ROBOSPECT European FP7 project [14], whose objective is that of building a complete autonomous robot able to collect images without human help and then process them using a convolutional network in order to identify the presence of a crack. The robot is very complex itself; it is composed of a standard mobile vehicle with an extended crane with a robotic arm. The computer vision system was composed of two CCD cameras and two DSLRs and a powerful lighting system mounted on the robotic arm. The robot also had a 3D laser scanner with the alleged ability to detect the depth of the crack. When the images are processed by the CNN and a crack is detected, the robotic arm is moved close to the crack and the 3D laser is activated for depth measurements.

While this project focused on one technology, other Non-Destructive Examination (NDE) technologies, including but not limited to LiDAR, Impact-Echo (IE) analysis, Thermal/InfraRed (IR) analysis, machine vision, might also have future promising research opportunities for concrete crack detection.

1.4. Autonomous Navigation

Later in this document we present details of design, development and implementation of two low-cost data acquisition systems for road tunnel inspections based on simple systems that do not require complex preliminary work and can be used also in tunnels with normal traffic flow [15], [16]. The prototype data acquisition systems were then used to generate a database of images from several tunnels in the Los Angeles county area and were subsequently used to train deep CNNs for automatic classification of acquired images and video. Indeed the second-generation data acquisition system developed in the 2017-2018 academic year [16], generated a series of high-resolution pictures used in a pre-trained Artificial Intelligence (AI) system based on a deep CNN (Inception-V4 pre-trained deep CNN from Google) able to identify the presence of cracks through the classification of the pictures achieving 98% accuracy. The next key problem to address for a fully autonomous detection system is autonomous navigation and annotation and tagging of the acquired images.

The problem of autonomous navigation relies on the ability of a vehicle to localize itself in an unknown environment, which is not an easy task indoors since the Global Positioning System (GPS) simply cannot work due to lack of visibility of the GPS satellites in a fully enclosed environment. While there is a multitude of augmentation systems one could envision for tagging certain locations within the tunnels or using beacons of various types, our focus is on methods that do not require ANY modification or augmentation system. For instance, the vast majority of tunnels throughout the world are unlikely to have any such augmentation systems and implementing and maintaining such a system could be a costly endeavor even if present.

One of the possible solutions to the problem of indoor localization in the absence of any augmentation system is Simultaneous Localization And Mapping or SLAM. SLAM techniques make possible the localization of an autonomous platform by mapping the environment in the first step and then localizing inside the environment through references called Landmarks. The principle of localization using SLAM is best described in terms of how humans localize themselves in an indoor environment. Much of our ability to localize ourselves indoors is based on using visual cues and identifying locations relative to stationary objects or landmarks within the environment. For instance, to guide a person to the food court inside a mall, a person may give directions based on the topology of the mall, name of stores along the path, or generic directions like “turn right after the information kiosk”. We can hardly orient ourselves based on for instance north-south orientation since even that requires using the sun as a visual cue. The two main sensor technologies exploited to implement SLAM are LiDAR and Camera or a mixed combination of the two.

Light Detection and Ranging (LIDAR) is an efficient method to scan the surrounding environment in two or even three dimensions [17]. The principle of operation is similar to RADAR which uses Radio Frequency (RF) waves that bounce from the surrounding environment and are picked up by the receiver and processed to determine the range to various obstacles. The main difference is operating frequency whereby LiDAR uses optical frequency range with a correspondingly higher resolution. Performing what is called Time-of-Flight (ToF) measurements, LiDAR calculates the time it takes for the light to reach the obstacle and return back to the emitter, in this way this technology senses the distances from objects occupying the space [18]. In the case of three-dimensional scanning, LiDAR can be used to build point cloud-based maps from which different object’s shapes can be recognized. Moreover, they are found to be useful in those applications requiring mapping under the surface, for example, deep crack detection or detection of surface degradations.

LiDAR provides a wealth of information that is not necessarily very relevant in so far as applications to localization are concerned. On the other hand, cameras seem to be very appealing for our purposes since they are cheap, light, compact and with low power consumption. Moreover, they are suitable for the detection of stable features and extraction of useful information from the environment (it is unlikely that features inside tunnels, for instance, change much over time) [19]. As far as localization is concerned, cameras, whether monocular or stereo, are at the base of the visual Simultaneous Localization and Mapping (vSLAM) techniques proposed in the literature. Images can be used to save the map of an unknown environment in order to globally position a vehicle or they can be used to track the motion of an autonomous system by subsequent reference frames transformations, each of them related to the previous state as the state evolves in time. The use of cameras is not limited just to localization, but they can also be exploited to have the perception of the depth and extract important information regarding obstacles or target objects. Finally yet importantly, from the images of the camera, the autonomous system can be enabled to discern between the nature of different objects using AI techniques and take decisions accordingly [20].

CHAPTER 2 – SYSTEM DESIGN, DEVELOPMENT AND IMPLEMENTATION

2.1. Video Based Data Acquisition

The basic architecture of the data acquisition system (parts of which we did not implement) is primarily composed of four core blocks. The first one is the video acquisition block. It is composed of three consumer grade cameras with their related LEDs array as lighting sources. This composition ensures a coverage of about 180° of the tunnel lining with the minimum equipment cost. All the devices are mounted on a steel framework supported by a vibration isolation pad that increases the performances of the image stabilization system of the cameras. Finally, a series of sensors like accelerometer, encoder, help in tracking, inside a tunnel, a possible detected crack.

The second block is part of the software and divides the input videos into their frames, which feed the model block. This is a pre-trained CNN that gives a prediction through the output layer. It is not needed in a real time evaluation and so, the computational power required is very low. Finally, the resulting classification and all data coming from the sensors are stored and organized by the last block of the system.

The low equipment cost, which is more than an order of magnitude cheaper than the industrial ones, the great portability and modularity are the key points of this module of the devised system presented here. It has a huge impact on the price and its quality greatly affects the accuracy of the model block. As a matter of fact, the capability of the neural network to identify cracks of small dimension is highly determined by the quality of the input images and of the dataset for the training session. The camera to be chosen, as well as the light source and the mobile base, should be carefully weighted trying to have a compromise between quality and price. Naturally, the presented prototype, is one of several possible options and only further field experimentation can really determine what is the best one.

Two major reasons has led to the choice of the design of the prototype-1. First, it had to be easy to install on a vehicle, remotely controlled within it and not annoying for other drivers. Secondly, it has been used to test and compare the acquisition system with an unusual source of light. The prototype-1 was made of four components: a consumer-grade camcorder, an infrared light torch, a remotely controlled pan tilt and a bicycle rack as base. Infrared lighting was selected in order to try a different spectrum of light, looking for advantages and disadvantages of it. The chosen infrared flashlight was the Evolva Future Technology T20 with a 38mm lens, emitting infrared light at 850nm and powered by one 18650 Li-Lon rechargeable battery (3.7-4.2V). Due to the chosen light source, a camera with a CCD sensor has been chosen.

CCD sensors are very sensible to infrared light but with the drawback to present a lower frame rate than CMOS sensors. Consumer-grade cameras all have a hot filter for infrared light, which must be removed to detect light in this spectral region. It is not simple to manually remove this filter but some cameras have a night vision mode that simply remove it mechanically turning the RGB image into a gray-scale one. The total cost of the prototype, shown in Figure 1, was less than $200. In order to have an order of magnitude, a leading manufacturer of industrial cameras, Imaging Source, has been contacted for a quotation. The cheapest camera without lens was around \$850, four times more expensive than the entire prototype.

Figure 1: Close-up picture of the prototype-1.

Figure 2: Prototype-1 mounted on the top of the vehicle.

2.2. High Resolution Image Based Data Acquisition System

Using what we learned from prototype-1, our second data acquisition prototype is composed of two main elements, the image acquisition system and the deep neural network. The first one, as previously explained, aims at collecting images for the analysis of the surfaces inside tunnels and generating a proper dataset for the training the network. The second element is a software program, which runs on a separate hardware and is used to train the Inception-v4 deep CNN for our purposes.

The idea behind this acquisition system is to devise a low-cost system that is, at the same time, easily reusable and does not require a complex infrastructure. In order to obtain acceptable results in the training of the network, a large number of high-resolution images must be acquired.

A very cheap video capture equipment operating in the infrared regime, from which (thanks to a post-processing phase) a series of pictures have been extracted, characterized the first generation prototype that was developed and tested in the 2016-2017 academic year. In the second-generation prototype presented here, the acquisition procedure has completely changed, and it is based on images acquired directly using a DSLR camera. We chose it thanks to the good results that this kind of cameras produced in several works presented in the literature and referenced in the previous sections. The other problem is clearly the tunnel inspection method, because the expected result is a system that has to work inside urban tunnels without disturbing the traffic flow to the extent possible. To accomplish this, the whole data acquisition system was envisaged to be a portable platform that could be easily mounted on the roof of any vehicle using roof racks. Once mounted, the electronic systems on the platform could be controlled from inside the vehicle, the main requirement being supplying power to the unit. The designed and developed system is composed of four main components: DSLR camera, lighting system, radio-frequency remote control and camera control unit.

The chosen DSLR camera is the CANON EOS800D model, with a 55mm lens. This model satisfies all the quality constraints required by our application. The 24MP picture resolution guarantees extremely good pictures, which can be taken at relatively high speed (up to nominal 6 frames per second or fps) and in different modes. Moreover, the manual mode of the camera allows the user to set all the parameters of the camera manually, in order to adapt the camera to the difficult environmental conditions of tunnels. Another important feature of the camera is the capability of being controlled remotely through its PC cable by an external CPU, which is crucial for the system in order to have a camera able to shoot without the human interference.

For this prototype we focused on the lighting system, which plays a very important role in this kind of application. Inside a tunnel, we usually have poor lighting conditions and so the camera must be assisted by external light source in order to produce clear pictures. Two light sources have been selected, one fixed light source and a camera-flash. The fixed light source is a 500-LED Panel (Neewer 500LED Photo Studio Lighting Panel), that is useful to create a strong and clear light. Synchronously, we also used the Neewer NW-561 LCD Display Speedlite Flash to generate a strong light source with the camera in order to improve the contrast on the pictures taken.

Moreover, we use radio frequency remote control in order to create a wireless connection between the camera and the flash mentioned above. In addition, it is essential to improve the scalability of the acquisition process; in a plausible expansion of the system with more cameras and flashes, they can be synchronized through this kind of control. PocketWizards offer a reasonable solution in terms of costs and quality. The company produces a variety of instruments for the flash and camera synchronization. A pair of PLUSIII units have been chosen in our system. One of them is connected to the camera through its own hotshoe, while the other is connected to the flash through a provided cable.

Finally, the camera control has to activate the shooting functions of the camera. For this purpose, a Raspberry PI3 has been selected, basically for the reduced dimensions which allow us to easily place it on the mounting platform and to connect it to the camera through a PC cable. The Raspberry is controlled through an external keyboard and a 7’’ LCD display. However, the main reason behind the choice of a Raspberry PI, is the capability to exploit the power of the Python library gphoto2. A simple Python code has been developed in order to activate the shutter release of the camera at its maximum speed (it is like having the shutter constantly held down by a finger).

Thanks to this technique, the camera produces pictures at the maximum speed allowed and the execution of the program can be easily stopped when needed.

The overall system is presented in Figure 3 and Figure 4, and adheres to all the constraints previously explained, with a total cost of below $2000. Everything is assembled on a homemade platform, which is composed of a wooden base and an aluminum-based frame that allows the entire platform to be tilted to take pictures from different angles, and steel camera holders.

Figure 3: The overall system composed of LED lighting system, DSLR camera, PLUSIII sync system and standalone flash, Raspberry PI, 7’’ LCD display for programming, and cables.

Figure 4: The system mounted on the minivan.

2.3. Camera Based vSLAM

A project has been carried out at Cal State LA in order to develop a fully autonomous localization system based on visual SLAM (vSLAM). There are three possibilities for the choice of the camera sensor and these are Monocular cameras, Stereo pair cameras, and time-of-flight cameras.

Monocular cameras or Monocameras, are sensors characterized by just one lens that capture frames providing no immediate and clear information about depth. Through the years, diverse techniques have been used to overcome this limitation of mono images. They can be coarsely classified into two main categories [21]:

- SFM: Structure-From-motion techniques exploit the physics of motion and perception to understand the distance of a moving object inside a fixed set of consecutive frames;
- DFC: Depth From Combining Defocus and correspondence techniques rely on different depth cues such as texture, Focus/Defocus, occlusion or gravity.

The main drawback of this approach aimed at converting two-dimensional images into three dimensions is that they are computationally expensive and not accurate enough for real-time, high frequency-based applications [22]. When monocular cameras are used for SLAM purposes, another approach is usually adopted to create depth perception and that is the acquisition of the same image frame from different points of view to exploit the principle of triangulation.

Time-of-Flight (ToF) cameras exploit the technology of time-of-flight measurement, just like LiDAR, and provide several pieces of information about the frame scene [23]. Basically, these cameras rely on the principles of light reflection:

- The illumination unit modulates a light source with a solid-state laser or LED operating near the infrared domain, at about 850nm;
- An imaging sensor which only responds to the same light frequency detects photons and converts energy into electric current;
- The phase shift between illuminating and reflected light is calculated and used to determine the distance from the camera plane of captured pixels [24].

The advantage of the ToF paradigm is its low software complexity, fast response time and independence from frame brightness. Moreover, range results can be scalable, from 10 to 103 meters. Nowadays ToF cameras have low-resolutions and integrated solutions are not common [25]. To overcome some of these problems, a possible solution is to use a regular RGB camera and a ToF sensor, and by calibrating the setup, a final 4-channel frame output can be provided (three-channel RGB plus the depth channel).

The term stereo camera refers to a hardware setup aimed at capturing two images of the same scene from two slightly different points of view in order to exploit the stereoscopic properties. Basically, a stereo system is made up of two identical lenses fixed at a certain distance called the baseline. Stereo cameras usage relies on the principle of Triangulation, which is the process able to determine the three-dimensional position of an object from multiple images, in this case, two. The way triangulation works is by calculating the disparity of the two images taken and determining the depth by using intrinsic parameters of the camera such as its focal length, distortion parameters, and baseline.

Given the basic overview presented above and considering the various tradeoffs, we have selected a stereo camera as our main sensor for the implementation of vSLAM. Besides the choice of the basic sensor, various visual SLAM techniques differ from each other based on how they work [19]. The first thing to investigate is the landmark position inside the map and to implement this feature, one must first identify the landmarks that need to be extracted from images of the environment under test. There are different approaches that can be adopted, the first relies on point features and it is the most commonly adopted technique. It exploits the Harris corner algorithm to detect the interest points and, as the autonomous system is moving, the SLAM algorithm updates the map with newly observed features, inserting new landmarks when required. Other, less used approaches include feature extraction from line or edges or even totally featureless algorithms. The first method works in a fashion similar to the point feature-based approach, except for the fact that edges (and not single points) are detected. It has the advantages of being light invariant and providing geometrical insight about the surrounding. Last but not least, a featureless approach is also plausible whereby information is extracted directly from subwindows of captured frames using pixel by pixel comparison.

Having discussed the generalities of feature extraction, it is now possible to explain how the localization system works. There are different techniques that can be exploited ranging from extended Kalman filtering or Particle filtering to key-frame based approaches. As the literature survey suggests [26], [27], nowadays SLAM techniques are focused on keyframe based algorithms and bundle adjustment (BA) local optimization. The way this approach works is similar to the Structure From Motion (SFM) technique, but it is not as computationally intensive. Harris corners detector are used to catch landmarks and then these are tracked among consecutive frames. Some of the frames used to track landmarks are selected as key-frames which are then used to build the map and localize the autonomous system [28]. The problem of tracking camera pose between two consecutive frames is referred to as the Perspective-n-Point problem (PnP) [29].

Given our investigation and preliminary studies, we selected the following hardware for the design, development, and implementation of vSLAM: ZED stereo camera as sensor and an NVIDIA Jetson TX2 massively parallel Graphics Processing Unit (GPU) as a single-board computer used for real-time processing of the acquired ZED images. In addition, two different libraries have been tested and compared in order to develop a SLAM system, ORBSLAM2 and ZED SDK. The ZED camera has been chosen also because it is the hardware used to implement the obstacle and collision avoidance software, a feature that can be deployed in an autonomous system. This choice avoids the use of additional hardware devices, that would result in additional cost, weight and integration effort. The satisfying performance of the camera combined with the above-mentioned advantages led us to prefer it over ToF cameras or LiDAR, whose slightly better performances related to this specific application do not outweigh the additional drawbacks. Figure 5 below depicts the basic configuration.

Figure 5: NVIDIA JETSON-TX2 massively parallel GPU and ZED stereo camera used for the implementation of vSLAM.

CHAPTER 3 – DATA SET GENERATION

3.1. Video Acquisition Systems

The prototype, shown previously, has been mounted on the roof of a minivan and carried through the major tunnels of Los Angeles. With the first data acquisition attempt, it has been possible to achieve a collection of videos lasting approximately 40 minutes (total time: 40.09, total number of frames: 2405.4 x 30 = 72162 frames). After an accurate selection, it has been possible to obtain a dataset of 6494 images.

- Crack detection folder: 4094
- No-Cracks detection folder: 2400
- Total dataset images: 6494

Finally, using an already introduced technique, artificially expanding the training data, through simple image modifications like cropping, rotating and scaling it has been possible to drastically increase the number of the available images. An example of the potentiality of this technique is presented in Figure 6.

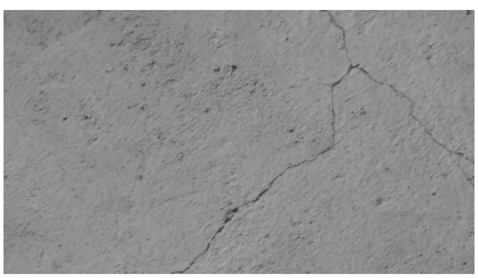


Figure 6: Practical example of artificially expanding the training on a crack image of dataset. The picture on the left is the one on the right randomly rotated of 209° counterclockwise.

The Model

The model algorithm is the third block of the system and differs with almost all related works present in literature. It exploits recent achievements in computer vision, brought by the application of Deep Learning. Indeed, this groundbreaking technique, all of a sudden, has exceeded all old IPTs and traditional machine learning algorithms, allowing, for this project, to use only images taken by low cost consumer-grade equipment. This has been made possible by the capabilities of convolutional neural networks to divide the problem in to simpler ones and to identify increasingly more complex patterns from the input images. After a model is trained on the desired features, it is simple to embed it in an actual working system. It will analyze input images coming from the pre-processing block giving a probability prediction for each class set.

Two Neural Network based methodologies have been implemented and trained on the specific task of crack detection with results presented later in this report. The results of the two approaches are proposed with different values of hyper-parameters such as learning rate, η, number of training epochs, regularization parameter λ , and mini-batch size.

3.2. Image Based Acquisition System

Our prototype has been designed to be able to capture a large number of pictures without loss of quality. To find the best parameter set, several tests have been performed in a dark room, so that the environmental conditions can reproduce the ones present during the real experiment. First, we consider the image-acquisition speed, which is set at the highest possible rate and represents a fixed constraint for data acquisition at normal driving speed of about 30 Km/hr. Our tests have shown that in order to extend the number of acquired images, it is better to set every parameter mode manually. In this way, it is possible to avoid the auto-setting of the camera, which introduces a small post-processing delay in the shooting time that is undesirable. Thanks to the powerful camera, we are able to capture more than one photo per second, while maintaining a good image resolution. Indeed, we were able to acquire at least 3 images/second driving at 30 km/h. This way, we could guarantee a proper number of good photos in the tunnel covering all regions of interest, without interfering with the traffic flow.

Three camera parameters have been modified: ISO speed (for controlling the light sensitivity of the camera), the aperture (for the background blur) and shutter speed (for objects in motion). The used setting is ISO3200, F4.5 aperture and 1/400 shutter speed. The focus of the lens has been left to the automatic procedure of the camera. During several tests directly performed in the tunnels, the automatic modes available on the camera did not respond well and not just in terms of speed of the shutter. For example, in sport mode (which at first thought may appear to be suitable for the application at hand) the camera produces pictures that are completely out of focus and cannot be used.

An example of one picture collected with the camera showing the extremely high resolution obtained is shown in Figure 7.

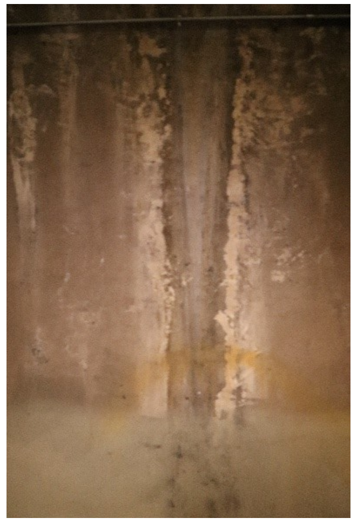

Figure 7: Example of a picture taken from the wall inside a tunnel obtained using the designed and developed prototype-2 system.

Inception-v4 and Transfer Learning

Transfer learning is a machine learning technique able to use a neural network in a specific application different from the one for which it has been pre-trained. A Convolutional Neural Network (CNN) is often structured in such a way that the first layers aim to find basic and common patterns (like edges, colors etc.) from a collection of training images, while the last layers effectively perform the classification functions for which the network has been designed for. In this way, through transfer learning, a user can use the learning power of a network until the last layers and retrain only the decisional layers to his/her specific application. In this paper, we used the Inception-v4 network which is the latest version of Google-Net, a powerful deep neural network presented by a Google team in the ImageNet Visual Recognition Challenge in 2012. Inception-v4 is a 48 layers deep network, based on the innovative concept of Network in Network structure, which has been trained by Google with powerful GPUs. This kind of deep network requires weeks for being well trained from scratch and requires a huge dataset to be sufficiently generalized to be used for a multitude of applications. Once the core network is trained however, thanks to transfer learning it is possible to retrain only the last layer of the network (i.e., the decisional layer), which is a not a very time consuming operation for reasonably sized datasets. Inception-v4 is a network which classifies images in certain classes provided by the user, so it is perfectly usable for the crack detection problem at hand. During the retraining process, it is possible to set different hyperparameters in order to adapt the network as much as possible to the provided dataset, for example adjustments can be made to the learning rate or the dimension of the different data batches.

3.3. Camera Based vSLAM

A sample of the stereo image generated by the ZED camera in the Engineering building at Cal State LA is depicted in Figure 8 below. The camera lens always has to be oriented forward during both the mapping and localization phases.

Figure 8: Stereo-pair images taken by ZED SDK.

ORBSLAM2 is a feature-based system, it extracts from frames key-point locations, while often inserting new keyframes and eliminating the redundant ones [26]. It is an Open Source software and hence, easily adaptable. It embeds a visualization system to see the map and trajectory during both the mapping and the localization phases, and it presents a minimal User Interface that makes it easy to swap between the two operational modes.

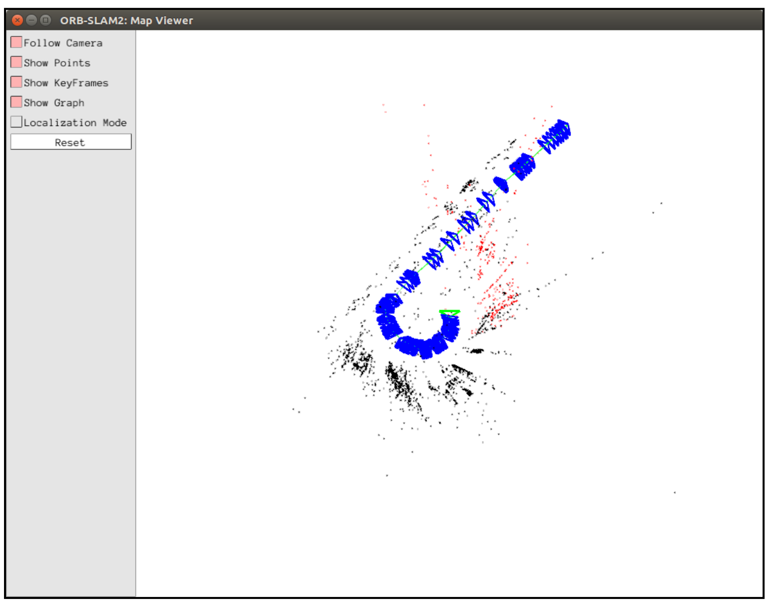

Figure 9: Map and trajectory visualization of ORBSLAM2.

It is possible to visualize in real-time the keyframe selection and map points generation and recognition, as depicted in Figure 9 above.

The other library tested for SLAM purposes has been ZED SDK. It is the library released by Stereolabs, the firm producing the ZED camera that has been integrated in the system. This is proprietary software and hence not modifiable, and no information about the software system architecture has been found in the literature. We used this library for comparison purposes. Both libraries exploit the CUDA parallel programming capabilities of the Jetson TX2 board, showing good graphical performances.

Figure 10: Mapping phase of ORBSLAM2 (performed on the third floor of the Engineering building at Cal. State LA). The green dots represent the points cloud of the environment.

Figure 11: Localization phase of ORBSLAM2 (performed on the third floor of the Engineering building at Cal. State LA). The blue dots represent landmarks.

ORBSLAM2 was found to be more easily accessible in the short term due to the graphical interface, but the lack of documentation makes it hard to effectively interact with hardware. On the other hand, ZED SDK runs natively on the stereo camera and provides low-hardware interaction, fair camera control, and really faster performance. During the tests on the vehicle, under equal circumstances, the graphic engine was running at 4-5 frames per second (FPS) with ORBSLAM2 while with ZED SDK, 60 FPS were achievable. The number of frames per second that can be processed in real-time is of fundamental importance for SLAM purposes, since tracking becomes more robust and fluid as FPS increases, resulting in less tracking losses. The mapping and localization phases inside the Engineering building of CSLA using ORBSLAM2 are depicted in Figure 10 and Figure 11 respectively.

One other observation is that ORBSLAM2 was not able to run without a previously charged map without losing track due to the low FPS and this constrained us to generate a map offline before the operation, while the ZED SDK performances made it possible to have online map generation, or even navigation without a map, exploiting only the visual odometry capability of the camera.

The speed of the stereo camera affects the results only when performing a turn; faster rotations yield lower accuracy. In straight paths, this parameter does not influence the results. Performing the test in a darker environment (or for instance, at night) would lead to the identification of a smaller number of landmarks, and thus, a poorly mapped environment. However, the accuracy reached during the localization phase with respect to the mapping phase would not change. This applies to both ambient lighting and vehicular supplied lighting.

CHAPTER 4 – RESULTS AND DISCUSSION

4.1. Video Based Data Acquisition

Re-trained CNN

Transfer Learning is becoming a very popular topic in machine learning community. Moreover, experimental evidence with Deep Learning techniques have demonstrated the possibility to successfully re-purpose an already trained deep convolutional neural network with novel generic tasks [30]. Indeed, all convolution and pooling layers extract increasingly abstract features that can be used to classify different types of objects. Instead, the fully connected layers and the classifier need to be re-trained on the new task, using supervised learning with the proper image database. In light of this, as first step of the model generation, we have taken a state of the art CNN and retrained it to detect the presence of cracks. In this way, it has been possible to train a huge model in less than one hour (a model that usually required 2-3 weeks to be fully trained).

The selected trained model is Inception-v4 model, a slimmed down version of the relative Inception-v3 model [31]. Unlike the other, this novel architecture has been designed specifically for TensorFlow library. It has shown very good performance at relatively low computational cost and in order to achieve these performances, it exploits inception blocks, batch normalization and residual connections in its sibling version Inception-ResNet-v2 [32].

The model has been retrained with input images of 299x299 pixels using different settings and hyper-parameters. Final results are presented in Table 1. The last training session has been performed using the artificially expanded dataset. The choice of hyper-parameters has been made taking into account common heuristic rules and the final accuracy has been computed with the test set, only presented at the last epoch. The test set was tuned at 10% of the dataset and containing 649 different pictures. The ninth test, that achieved 98.1% of test accuracy, classified in the correct class 637 images with only 12 misplaced. This result demonstrates the huge potential of transfer learning that already with the first simulations has proven to be able to reach high values of accuracy.

Table 1: Transfer Learning results applying different hyper-parameters.

Test Number	Iterations	Learning Rate	Train Batch Size	Cross Entropy	Test Accuracy
1	4000	0.01	100	0.143	95%
2	9000	0.01	100	0.109	97.5%
3	9000	0.001	100	0.132	96.4%
4	15000	0.001	100	0.147	95.1%
5	15000	0.01	100	0.103	97.6%
6	10000	0.01	100	N.A.	N.A.
7	9000	0.01	300	0.102	97.6%
8	9000	0.01	1000	0.106	97.5%
9	9000	0.01	1000	0.094	98.1%

Custom CNN

A custom convolutional neural network, despite the lack of a large dataset, has been trained using supervised learning. Unlike Inception-v4, a CNN trained from scratch on a small number of classes is more suitable for the specific assigned task. Indeed, all weights and biases are specifically calibrated to recognize features of the selected classes. The definition of the network architecture does not have a closed solution. Many different approaches can be applied and usually, only the actual simulation can discern what is the most suitable framework. For this project, all decisions have been made trying to keep the number of parameters low. A high number of parameters not only increases the training process time but also makes the network more inclined to overfit input data. Figure 12 gives a detailed overview of the designed architecture. It has two similar stages of convolution and pooling and a final soft-max classifier that has as output a certain prediction over two different classes. Going deeper with the network, the number of parameters of a single feature decreases and after the second pooling the output matrix takes the shape of an array in order to be suitable for the fully connected layer.

The input layer has output with dimension 128x128 pixels for each color channel. Images are in gray scale but the color model is RGB. First convolutional layer (CL), like the second one, has a 5x5 filter with zero padding in order to analyze presence of cracks in the entire picture. Then, if a crack is detected, it is not important anymore its position within the image so that a max pooling (PL) is exploited to decrease the number of parameters. Finally, fully connected layers (FCL) analyze the extracted features and the soft-max output layer generates a prediction. All neurons, except the ones of the last layer (sigmoid neurons), are rectified linear units (ReLu). This type of neurons have proved to be easier and faster to train [33]. Table 2 presents an overview of the designed convolutional neural network.

This is only one possible architecture. Many other frameworks can be devised and, as it is for hyper-parameters, only empirical and not deterministic rules are available for the design process.

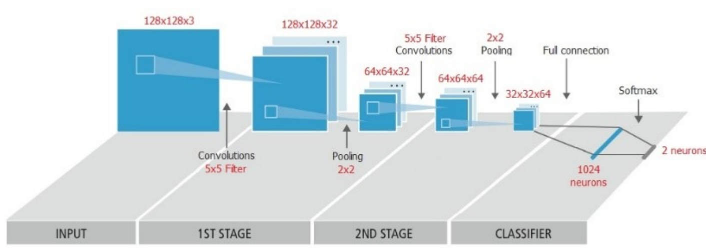

Figure 12: Architecture layout of the convolutional neural network.

Table 2: Summary table of the network specifications.

Section	Input	Output	Patch	Strike	Filters
Input Layer	1920x1080x3	128x128	N.A.	N.A.	3
1^{th} CL	128x128x3	128x128	5x5	1	32
1^{th} PL	128x128x32	64x64	2x2	1	32
2^{nd} CL	64x64x32	64x64	5x5	1	64
2^{nd} PL	64x646x64	32x32	2x2	1	64
FCL	64x64x32	1024	N.A.	N.A.	N.A.
Softmax Layer	1024	2	N.A.	N.A.	N.A.

In order to compare the results achieved with this model with the ones obtained with Inception-v4 model, we selected an equal size of the test set (10%) and the same percentage of validation images (20%). In every simulation, the validation occurs every 150 steps.

Unfortunately, it was not possible to maintain the same input dimension of the pictures due to a lack of the HW setting. Surprisingly, with only input images of 128px per side, it was possible to achieve 93.1% of test accuracy. So, it is highly probable that greater dimension of the input images, carrying more information, could greatly help the network learn more robust and useful features with a resulting higher accuracy.

Again, the simulations have been carried out with different combination of hyper-parameters and only the ones of the last attempt are reported in Table 3. The last simulation has been performed with 10000 steps and Dropout, L2 regularization and artificially expanding the training data in order to tackle the overfitting problem. Finally, the model has been trained with GPUs in order to speed up the entire process.

Table 3: Summary of hyper-parameters selected for the last simulation.

Image Size	Hyper-parameters				
	Train batch	***Validation b.***	***Dropout***	***L. rate***	***L2 reg.***
128px	150	200	0.5	0.00001	10

Comparison between the two models

Both networks are easily implementable. After the training session it is possible to store the model in a binary format file. This file can be loaded and used in the software of the device in order to make predictions about new input images.

Further works and improvements are needed, but the early results have been extremely promising. As expected, transfer learning, which needs less data for its learning process, has overtaken the custom neural network in terms of learning time and test accuracy. Considering the huge number of parameters that had to be trained, the result of 93.1% achieved by the custom CNN has to be considered as great. Little adjustments of the network, but especially a larger dataset and greater dimension of input images should increase the accuracy level around the value achieved by

Inception-v4. Indeed, the graph presented in Figure 13 shows that, due to the dimension of the image database, the custom deep neural network, after 7000 iterations, starts to overfit the data.

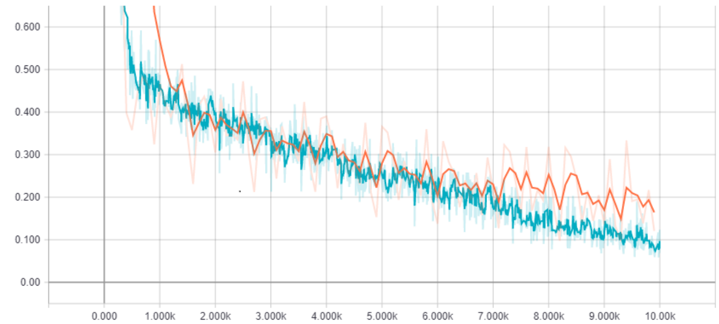

Figure 13: Cross Entropy functions. Blue line is the training function and red line is the validation function, smoothing=0.3.

4.2. Image Based Data Acquisition

Table 4 shows the results obtained by the retraining process with several choices of hyperparameters. The tests have been performed with the dataset generated by using the second data acquisition prototype based on DSLR camera. The dataset has been divided into a training set (80% of the images), validation set (10% of images), and test set (10% of images). The results have been evaluated in terms of test accuracy (the percentage of correctly classified images seen for the first time after the training) and cross entropy value on the last training sample, since the cost function used for the learning procedure is the cross entropy between the expected and obtained result. The network works with images with resolution of 299x299 pixels.

Table 4: Test Results.

Test Number	Iterations	Learning Factor	Train Batch Size	Validation Batch Size	Evaluation Interval	Cross Entropy	Test Accuracy
1	4500	0.1	100	202	100	0.002974	92.7%
2	2000	0.1	100	100	100	0.002808	93.6%
3	1000	0.1	100	100	100	0.009057	91.7%
4	2500	0.01	100	100	100	0.02428	93.6%
5	6000	0.001	100	100	100	0.07742	94.5%
6	9000	0.001	100	100	100	0.104863	93.6%
7	3000	0.1	50	100	100	0.002662	91.7%
8	2000	0.1	75	100	100	0.003804	92.7%

9	2000	0.1	150	100	100	0.006381	92.7%
10	9000	0.001	50	100	100	0.055273	93.6%

The software has been run on a HPZ840 Workstastion, which uses an Intel Xeon CPU processor at 2.2GHz and 16GB of RAM.

The current dataset is not large enough for this kind of application, nonetheless the detection accuracies are encouraging. Moreover, applications involving image processing can be executed much faster when run on a GPU, which has not been used in this first development stage.

Note that a test accuracy of 97.8% has been achieved in Test #7, which is a very good result for the current structure of the network and the current equipment; this means that the network misclassified (i.e., did not detect a crack when there was one or detected a crack that was not present) in only 11 of the 512 tested images. Out of these 11 misclassifications, 9 were "false positives" (i.e. a crack was detected were there was not one) and 2 were "false negatives" (i.e. the presence of a crack was missed by the algorithm).

Some results are also affected by overfitting, which may occur in presence of a small dataset. Overfitting is a bad condition during the training phase and it arises when the network starts to remember the training images "by heart" and effectively stops to learn. A clear example is shown in Figure 14, which refers to Test #1, where it is possible to appreciate how after 2000 iterations the training accuracy (orange line) remains constant and equal to 100%, which means that the network learned only the specific aspects of the training images and not the general patterns of the presence of a crack. In fact, the Test #1 validation accuracy is effectively far from 100%. The best way to overcome the overfitting problem is to enlarge the dataset or to modify heuristically the hyperparameters, but different tests we conducted has verified our belief that the big problem is effectively the size of the dataset. One possible way to handle this problem can be the artificial expansion of the data, obtained by rotating or cropping the images in order to increase the number of available pictures. Unfortunately, the current equipment is too slow for artificial data expansion and a simple retraining can take weeks, losing completely the good aspects of the transfer learning technique.

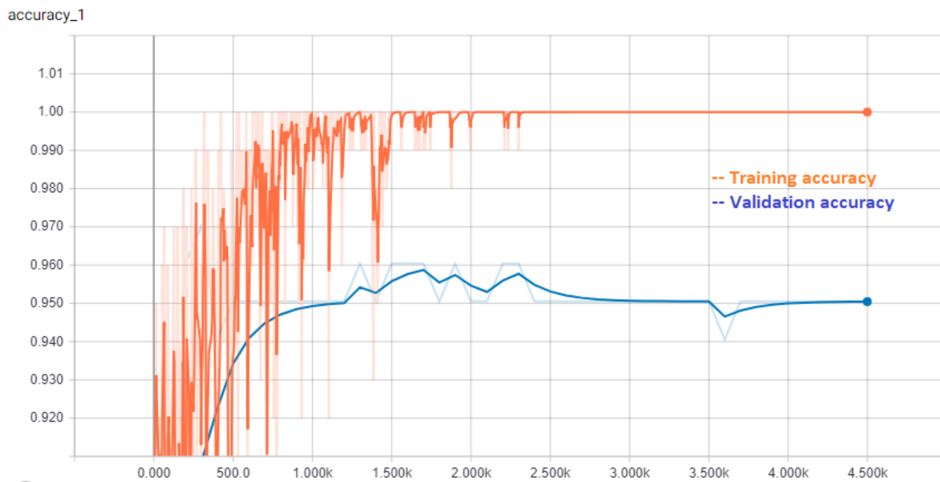

Figure 14: Accuracies on Test #1. Horizontal axis is the neural network training iteration index, while the vertical axis is the classification accuracy on scale of 0 to 1, with 1 being 100%.

4.3. Camera Based vSLAM

Figure 10 depicts the result of the mapping phase of the third floor of the Engineering building at Cal. State LA generated by ORBSLAM2, where the green dots represent the points cloud. Figure 11 depicts the localization phase of ORBSLAM2 on the third floor of the Engineering building at Cal. State LA, where the blue dots represent landmarks.

Localization Accuracy

In order to assess the quality of the localization system based on vSLAM, we conducted several experiments focused on understanding how precisely can a vehicle localize itself in an already mapped environment. The software developed for SLAM purposes exploits the ORBSLAM2 library in order to build a map of the environment and then localize the system in the mapped environment. In our experiment, a ZED stereo camera has been used to record two different video sequences of an L-shaped path inside the Engineering and Technology building of Cal State LA. Each sequence has been analyzed once to build a map and, in a second phase, three different localization tests have been performed on each, for a total of six trials. The platform was moving at 1 m/s during all the trials. The first sequence is composed of 982 frames taken at 30 FPS, while the second is composed of 1039 frames, again taken at 30 FPS. Mapping of the environment took 4.5 min for sequence-1 and 5.1 min for sequence-2. It is worth noticing that no true information about trajectory was available during the tests, and because of this, the vector of camera poses recorded during the mapping phase has been assumed as the true estimation. The just mentioned pose has been compared to the measurements taken during the evaluation attempts of the localization trials. We should point out that this assumption does not lead to a loss of generality since even if the map reference frame is not coincident with the world reference frame, the localization technique would still serve the purpose of localizing inside the mapped environment. Landmarks and features are stored inside the map and they are assumed to be stationary in space, and indeed a relative localization with reference to them is all that is needed in annotating the pictures taken from the interior of the tunnels or other UTI. The results of our tests are reported in Table 5 below. The mean errors related to the trajectory of the platform during the localization phase are computed with respect to the pose of the camera during the mapping phase for all the trials performed. They are reported both in relative terms (percentage) and in absolute terms (millimeters), regarding both the axis of motion X and Z.

Table 5: Localization relative accuracy tests. The mean relative errors in the X and Z directions (in the plane of motion) are reported as both percentage values and as values in mm. L denotes the number of lost tracks (zero in all cases) and σ^2 denotes the error variance.

Seq.1:		$\mu_{relerrx}$	$\mu_{relerrz}$	μ_{errx}	μ_{errz}	σ_x^2	σ_z^2	L
• 982 Frames	Trial1	2.02%	1.53%	0.6 mm	1.6 mm	1.3 E-4	1.9 E-4	0
• Map. in 4.5 min	Trial2	2.21%	1.17%	0.7 mm	1.7 mm	1.2 E-4	2 E-4	0
• 30 FPS	Trial3	2.49%	0.83%	0.9 mm	2.1 mm	1.4 E-4	1.9 E-4	0
Seq.2:		$\mu_{relerrx}$	$\mu_{relerrz}$	μ_{errx}	μ_{errz}	σ_x^2	σ_z^2	L
• 1039 Frames	Trial1	0.67%	1.56%	0.8 mm	2.2 mm	8 E-5	1.7 E-4	0
• Map. in 5.1 min	Trial2	2.32%	2.11%	1.2 mm	1.3 mm	9 E-5	1.7 E-4	0
• 30 FPS	Trial3	2.27%	1.76%	1.6 mm	0.13 mm	7.8 E-5	1.5 E-4	0

From the measurements of variances, it is possible to notice that repeatability of the localization phase permits reliable operations since the variances remain fairly constant among different attempts. Figure 15 is the plot of the trajectories for sequence-1 and -2 showing repeatability for different data sets. The axes represent the X and Z directions (the plane of motion), and the unit of the axes is meter. Figure 15 displays the path that the platform followed during the trials whose mean errors and variances are shown in Table 5. Each triangle represents the position of the platform at fixed time intervals (of 1.7 seconds) for each localization trial in both the sequences, while the straight lines represent the camera pose during the mapping phase. As far as mean localization error is concerned, values are in the order of millimeters; however, we must point out that this error does not represent the absolute position estimate value but may be viewed as a result of the noise perturbing a previously built map.

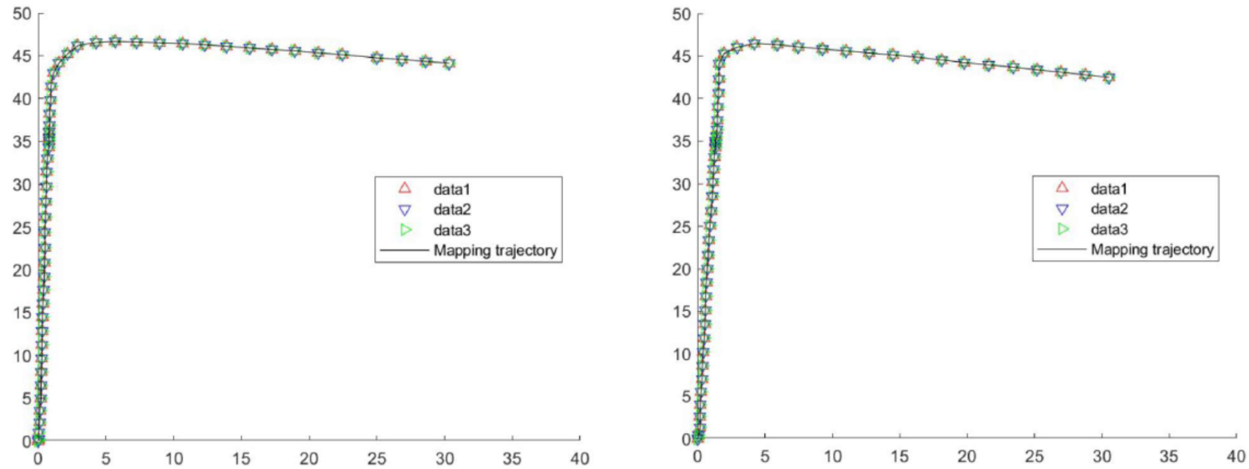

Figure 15: Plotted trajectories of sequence-1 (left) and sequence-2 (right).

To validate the system for operation in tunnels we carried a series of experiments in the Griffith park tunnel in the Los Angeles County area. In this case, the platform was moving at 20 mph. The results of the successful operation of vSLAM in this tunnel are depicted in Figure 16 and Figure 17 below.

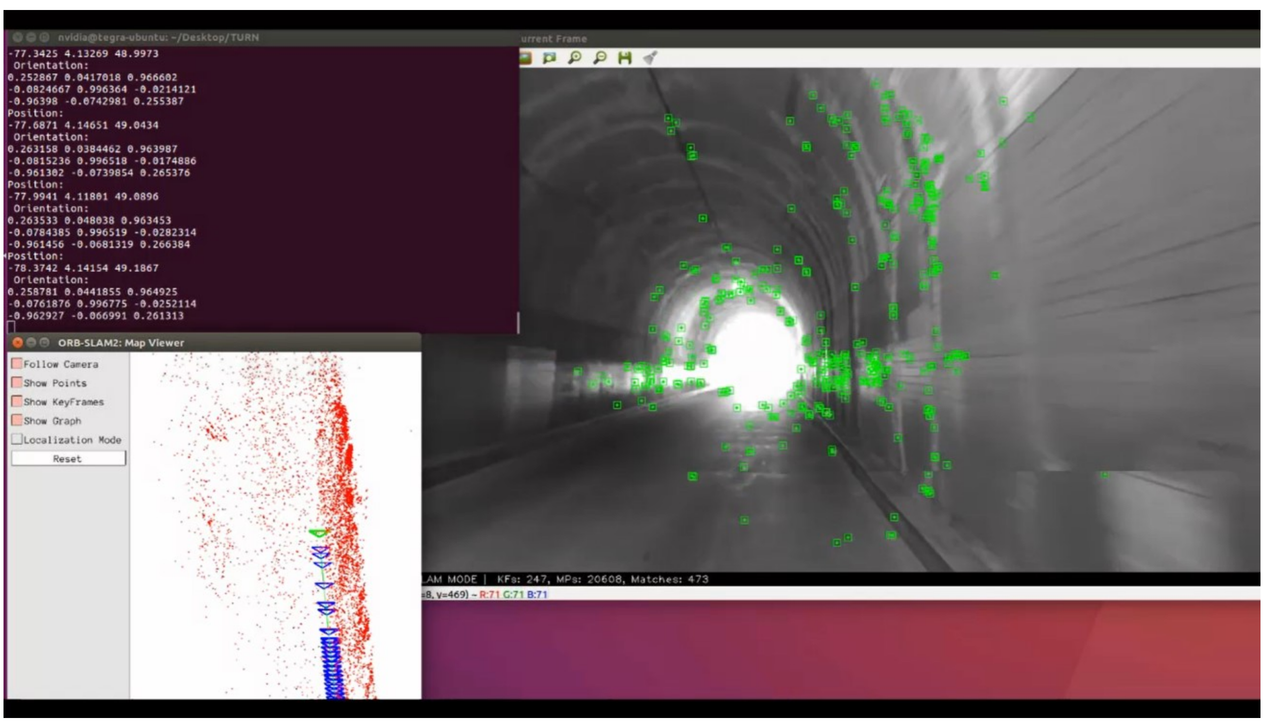

Figure 16: Operation of vSLAM during the map generation in the Griffith park tunnel in Los Angeles County. Reference cloud points are marked as green squares in the image above on the right. In the lower-left corner, blue triangles reflect landmarks identified

during the map generation. Note that the right-hand boundary of the tunnel is clearly defined.

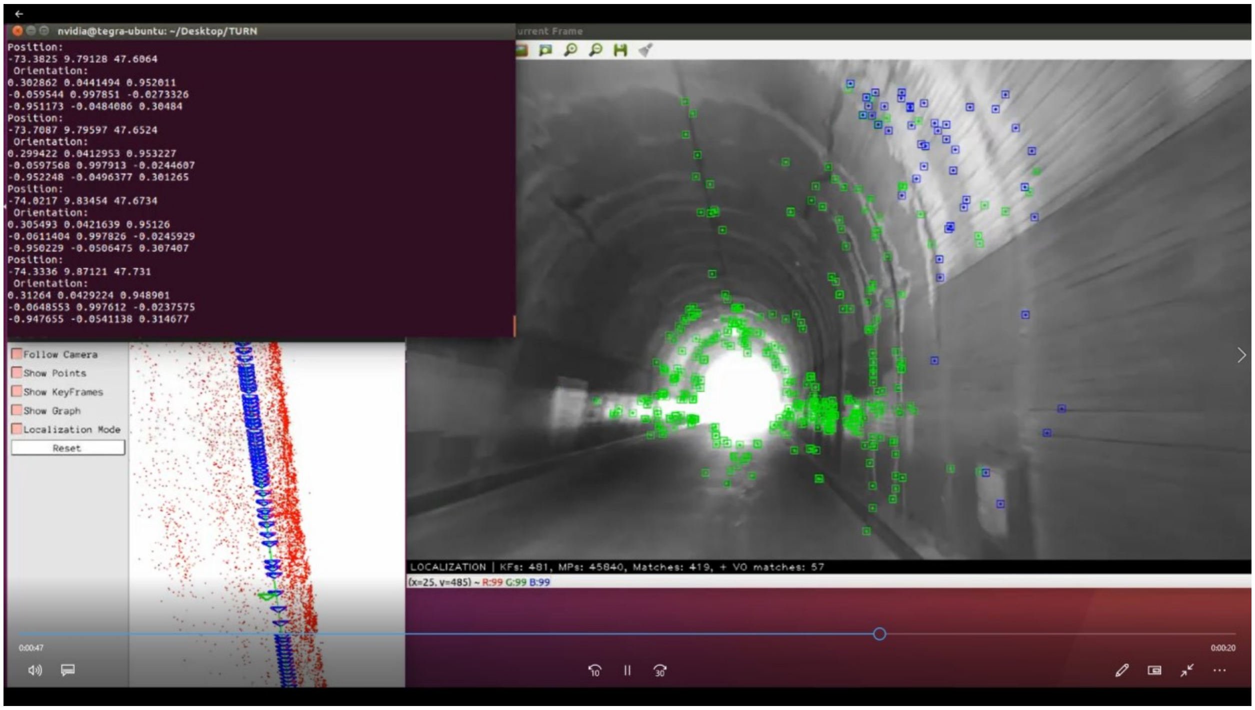

Figure 17: Operation of vSLAM during the localization phase in the Griffith park Tunnel in Los Angeles County. Once the map has been generated, the actual location of the vehicle (marked as a green triangle on the lower left square) relative to the previously mapped trajectory (marked as a sequence of blue triangles on the lower-left square) can be determined.

At the time the tests were performed, it was not possible to obtain specific permits to run multiple trials at fixed conditions in such a crucial and crowded location; hence, only the right-hand side of the tunnel could be properly identified as the right-hand lane was the only one viable. Performing the test at the exact center of the road would have led to symmetric identification of the tunnel sides, allowing to process the results like the ones obtained at Cal State LA. Nevertheless, it is remarkable how the platform was able to localize itself even under these conditions.

CHAPTER 5 – SUMMARY AND CONCLUSION

In connection with video based data acquisition system, a low cost, easily enforceable automatic cracks detection system, based on recent developments in image recognition and computer vision, was developed as a replacement to the commonly used manual inspection. Results have pointed out the suitability of deep learning architectures for the tunnel defect inspection problem relying only on low range, consumer grade equipment. Further works and researches may automate on a large scale the current tedious and unreliable work of underground infrastructures assessment.

Both custom CNN and retrained deep CNN tested have successfully proven that convolutional neural networks, with a small effort, can overcome most of the traditionally complex and accurately calibrated image processing techniques. Moreover, libraries like TensorFlow have drastically simplified network design and implementation and, their tools and features have gradually opened the possibility to train deeper and more precise neural networks.

In connection with image based data acquisition system, we have described the design, development and testing of a low-cost prototype for automatic crack detection on surfaces inside road tunnels using DSLR cameras. The first results produced are effectively promising and we obtained a 94.5% test accuracy with a small dataset and without a GPU. Moreover, our outcomes demonstrate how transfer learning could produce good results in tackling the crack detection problem. Finally, this relatively low-cost prototype produces high-quality images at normal driving speeds, provided the parameter set of the DSLR camera is properly adjusted.

In connection with the autonomous localization problem using vSLAM, we have presented the results of our preliminary investigation into localization and navigation techniques for indoor environments in the absence of any GPS signals or augmentation systems. The speed of operation of the localization system is of the essence for proper tagging of the acquired images which are obtained in real-time from a moving vehicle traveling through with normal traffic. Thus, we have implemented vSLAM on a massively parallel GPU (NVIDIA Jetson TX2). We have verified the basic functionalities of the real-time localization system using two different software libraries with various speeds in terms of the number of frames per second that can be processed. Future research might involve the implementation of SLAM through a 360-degree bi-dimensional LiDAR integrated into the already deployed vSLAM with the aim of exploiting the benefits of both methods.

As a final consideration, we observe that autonomous robotic platforms are being developed at break-neck speed. In not so far in the foreseeable future, such platforms equipped with a multitude of sensors will be able to perform inspections of the very vast transportation infrastructure anywhere in the world. As with many other projected applications of autonomous systems, there is going to be a long period of validation and certification whereby human operators need to supervise the operation of the platforms to ensure they perform as they should. But, in our humble opinion, one does not need a 100% functional system to reap the benefits of the technology. In this sense, in the intermediate-term, we believe that autonomous systems technology is yet another tool in the arsenal of the personnel dealing with the maintenance of the transportation infrastructure.

REFERENCES

[1] D. o. T. Federal Highway Administration, «National Tunnel Inspection Standards (NTIS): 80 Federal Register 41350,» 2015.

[2] T. Yamaguchi, S. Nakamura e S. Hashimoto, «An efficient crack detection method using percolation-based image processing,» *Industrial Electronics and Applications, 2008. ICIEA 2008. 3rd IEEE Conference on,* 2008.

[3] D. Qi, Y. Liu, X. Wu e Z. Zhang, «An algorithm to detect the crack in the tunnel based on the image processing,» in *Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2014 Tenth International Conference on*, 2014.

[4] S. Stent, R. Gherardi, B. Stenger, K. Soga e R. Cipolla, «Visual change detection on tunnel linings,» *Machine Vision and Applications,* vol. 27, n. 3, pp. 319-330, 2016.

[5] J. Laurent, R. Fox-Ivey, F. Sanchez Dominguez, J. Antonio e R. Garcia, «Use of 3D Scanning Technology for Automated Inspection of Tunnels,» *the World Tunnel Congress,* 2014.

[6] Z. Liu, S. A. Suandi, T. Ohashi e T. Ejima, «Tunnel crack detection and classification system based on image processing,» in *Machine Vision Applications in Industrial Inspection X*, 2002.

[7] S. A. I. Stent, C. Long, P. J. G. Girerd e R. Cipolla, «A Low-Cost Robotic System for the Efficient Visual Inspection of Tunnels,» in *Proceedings of the International Symposium on Automation and Robotics in Construction (ISARC)*, 2015.

[8] E. Protopapadakis, K. Makantasis, G. Kopsiaftis, N. Doulamis e A. Amditis, «Crack Identification Via User Feedback, Convolutional Neural Networks and Laser Scanners for Tunnel Infrastructures.,» in *VISIGRAPP (4: VISAPP)*, 2016.

[9] S.-N. Yu, J.-H. Jang e C.-S. Han, «Auto inspection system using a mobile robot for detecting concrete cracks in a tunnel,» *Automation in Construction,* vol. 16, n. 3, pp. 255-261, 2007.

[10] C.-H. Lee, Y.-C. Chiu, T.-T. Wang e T.-H. Huang, «Application and validation of simple image-mosaic technology for interpreting cracks on tunnel lining,» *Tunnelling and Underground Space Technology,* vol. 34, pp. 61-72, 2013.

[11] S. Gavillan, «Mobile Inspections system for high resolution assessment of tunnels,» in *The 6th International Conference on Structural Health Monitoring of Intelligent Infrastructure*, Honk Kong, 2013.

[12] K. Makantasis, E. Protopapadakis, A. Doulamis, N. Doulamis e C. Loupos, «Deep convolutional neural networks for efficient vision based tunnel inspection,» in *Intelligent Computer Communication and Processing (ICCP), 2015 IEEE International Conference on*, 2015.

[13] Y.-J. Cha, W. Choi e O. Büyüköztürk, «Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks,» *Computer-Aided Civil and Infrastructure Engineering,* vol. 32, n. 5, pp. 361-378, 2017.

[14] R. Montero, E. Menendez, J. G. Victores e C. Balaguer, «Intelligent robotic system for autonomous crack detection and caracterization in concrete tunnels,» in *Autonomous Robot Systems and Competitions (ICARSC), 2017 IEEE International Conference on*, 2017.

[15] V. Mazzia, F. Daneshgaran e M. Mondin, «Use Of Deep Learning For Automatic Detection Of Cracks In Tunnels,» in *29th Italian Workshop on Neural Nets. Recent Advances of Neural Network Models and Applications (WIRN 2019)*, Vietri sul Mare (Salerno, Italy), 2019.

[16] F. Daneshgaran, L. Zacheo, F. D. Stasio e M. Mondin, «Use of Deep Learning for Automatic Detection of Cracks in Tunnels: Prototype-2 Developed in the 2017–2018 Time Period,» *Transportation Research Record,* May 2019.

[17] W. Hess, D. Kohler, H. Rapp e D. Andor, «Real-time loop closure in 2d LiDAR SLAM,» in *Proceedings of the 2016 IEEE International Conference on Robotics and Automation*, Stockholm (Sweden), 2016.

[18] B. Schwarz, «Mapping the world in 3D,» *Nature Photonics,* vol. 4, 2010.

[19] M. Naveed, D. Fofi e S. Ainouz, «Current state of the art of vision-based SLAM,» in *Image Processing: Machine Vision Applications II, Vol 7251*, 2009.

[20] J. Redmon, S. Divvala, R. Girshick e A. Farhadi, «J. Redmon; S. Divvala; R. Girshick; A. Farhadi,» in *Proceedings of the 2016 IEEE International Conference on Computer Vision and Pattern Recognition*, 2016.

[21] P. Li, D. Farin, R. K. Gunnewiek e P. d. With, «On creating depth maps from monoscopic video using structure from motion,» in *Proceedings of the 2006 IEEE Workshop on Content Generation and Coding for 3D-television*, Eindhoven, 2006.

[22] Q. Wei, J. Shang, Z. Chao e Z. Zhenpu, «A real-time 2d to 3d video conversion algorithm based on image shear transformation,» in *Proceedings of the Fifth International Conference on Instrumentation and Measurements, Computer, Communication and Control*, Qinhuangdao, 2015.

[23] V. Ganapathi, C. Plagemann, D. Koller e S. Thrun, «Real-time motion capture using a single time-of-flight camera,» in *Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010.

[24] V. Castaneda, D. Mateus e N. Navab, «SLAM combining ToF and high-resolution cameras,» in *Proceedings of the IEEE Workshop on Applications of Computer Vision (WACV)*, 2011.

[25] L. Li, «Time-of-flight camera - an introduction,» Technical white paper for Texas instrument, 2014.

[26] R. Mur-Artal e J. D. Tardos, «ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras,» *IEEE Transactions on Robotics,* 2017.

[27] H. Strasdat, J. Montiel e A. Davison, «Visual SLAM: Why filter?,» *Image and Vision Computing,* vol. 30, n. 2, p. 65–77, 2012.

[28] G. Klein e D. Murray, «Improving the agility of Keyframe-Based SLAM,» in *Forsyth D., Torr P., Zisserman A. (eds). Computer Vision – ECCV 2008. Lecture Notes in Computer Science, Vol 5303*, Berlin, Heidelberg. , Springer, 2008.

[29] V. Lepetit, F. Moreno-Noguer e P. Fua, «EPnP: An Accurate O(n) Solution to the PnP Problem,» *International Journal of Computer Vision,* 2009.

[30] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng e T. Darrell, «Decaf: A deep convolutional activation feature for generic visual recognition,» in *International conference on machine learning*, 2014.

[31] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens e Z. Wojna, «Rethinking the inception architecture for computer vision,» in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[32] C. Szegedy, S. Ioffe, V. Vanhoucke e A. A. Alemi, «Inception-v4, inception-resnet and the impact of residual connections on learning.,» in *AAAI*, 2017.

[33] V. Nair e G. E. Hinton, «Rectified Linear Units Improve Restricted Boltzmann Machines,» *Proceedings of the 27th International Conference on Machine Learning,* 2010.

APPENDIX A – TECHNOLOGY TRANSFER ACTIVITIES

1 Accomplishments

Vehicle based data acquisition systems (camera and video based) were designed, developed, implemented and tested using Commercial Off The shelf equipment (COTs) in years 2017 and 2018. The deep convolutional neural network (CNN) was trained based on the acquired image database achieving 98% accuracy in automated detection of cracks and other blemishes. Acquired images need to be tagged with location information within the tunnel and annotated. In 2019, multiple approaches were studied for localization within tunnels and other underground infrastructure in the absence of any GPS, WiFi or other beacon signals which may be used to define reference locations. An indoor localization technique using stereo cameras named visual Simultaneous Localization And Mapping (vSLAM) was designed, developed, implemented and tested in 2019.

1.1 What was done? What was learned?

1) We have designed, developed and build two low cost data acquisition platforms, one video based and the other image based for taking video or images of the interior lining of tunnels.
2) We have tested the units in several tunnels in Los Angeles area and generated a database of images and video for further processing.
3) We used the databases generated to train and test a deep Convolutional Neural Network (CNN) that has been pre-trained by Google, namely, Inception-v4. We achieved excellent results with over 98% detection accuracy.
4) We designed and trained a fully custom NN with all parameters adjusted by our team and tested the unit on the generated database, again with excellent results.
5) We designed, developed and build a visual Simultaneous Localization and Mapping (vSLAM) system for localization in indoor environments in absence of any GPS or beacon signals. We then tested the system both in the Engineering building and in Griffith park Tunnel of Los Angeles with excellent results. The unit can be used for data annotation in a fully autonomous crack detection system.
6) We learned that one does not need sophisticated equipment to detect the most common types of cracks in Tunnels. More complicated inspection and analysis can be conducted after initial crack detection should it become necessary.
7) We learned that deep CNNs do an outstanding job of automated data classification provided they are properly trained. Much more extensive tests are needed using a vastly larger database of images to be able to extend the system for commercial use.
8) Several issues that are outside the scope of our research need further investigation; these include: a) platform stability and vibration tolerance while in motion, b) how to provide very strong lighting which is key to obtaining excellent and sharp images, c) how to artificially enlarge the image database since avoiding anomalous behavior of the CNN requires training on an extremely large database.
9) We have started exploring development of autonomous robotic platforms that could be used for tunnel inspection when armed with a suitable set of sensors. Collaborative

autonomous fleet of robots could have significant benefits in infrastructure monitoring applications.

1.2 How have the results been disseminated?

1) Annual meetings at lead institution
2) Conference presentations
3) Journal publications
4) Development of machine learning course at CSLA
5) Students' thesis

2 Participants and Collaborating Organizations

Names: F. Daneshgaran, and M. Mondin
Location: California State University, Los Angeles (CSLA)
Contribution: Study conception and design; system architecture development; student advisement and guidance; data analytics and Machine learning; interpretation of results; manuscript and publication preparation

Video Based Data Acquisition System:
Names: Vittorio Mazzia, and Marco Forestello
Location: CSLA
Contribution: System assembly and testing; data collection; training of deep CNN; analysis and interpretation of results; report writing; conference presentation

Image Based Prototype Data Acquisition System:
Name: Luca Zacheo, and Francesco Di Stasio
Location: CSLA
Contribution: System assembly and testing; data collection; training of deep CNN; report writing; analysis and interpretation of results

Visual Simultaneous Localization and Mapping:
Name: Antonio Marangi, Nicola Bruno, Fausto Lizzio, and Khashayar Olia
Location: CSLA
Contribution: System assembly and testing; data collection; report writing; analysis and interpretation of results

3 Outputs

Journal publications

[1] F. Daneshgaran, A. Marangi, N. Bruno, F. Lizzio, M. Mondin , K. Olia, "Visual Simultaneous Localization and Mapping with Application to Monitoring of Underground Transportation Infrastructure", Revised for publication in Transportation Research Record (TRR), paper 20-05103, March 2020.

[2] F. Daneshgaran, L. Zacheo, F. Di Stasio, M. Mondin, "Use of Deep Learning for Automatic Detection of Cracks in Tunnels: Prototype-2 Developed in the 2017–2018 Time Period", TRB Journal, https://doi.org/10.1177/0361198119845656, May 3, 2019.

[3] V. Mazzia, F. Daneshgaran, and M. Mondin, "Use of Deep Learning for Automatic Detection of Cracks in Tunnels," in Progresses in Artificial Intelligence and Neural Systems, pp. 99-101, Springer Book series, 2019.

Presentations

[1] A. Marangi, F. Daneshgaran, N. Bruno, F. Lizzio, M. Mondin, K. Olia, "Visual Simultaneous Localization and Mapping with Applications to Monitoring of Underground Transportation Infrastructure", Transportation Research Board Annual Meeting, January 12–16, 2020, Washington D.C.

[2] A. Marangi, K. Olia, F. Daneshgaran, N. Bruno, F. Lizzio, M. Mondin, "Simultaneous Localization and Mapping with Application to Monitoring of Underground Transportation Infrastructure", International Symposium on Advanced Electrical and Communication Technologies ISAECT2019 that will take place at the University of Rome Tor Vergata, Rome, Italy, from 27th to 29th November 2019.

[3] V. Mazzia, F. Daneshgaran, M. Marina, "Use of Deep Learning for Automatic Detection of Cracks in Tunnels", 29th Italian Workshop on Neural Networks (WIRN), Vietri sulMare, Italy, June 2019

Major reports

1) *Master's thesis: Vittorio Mazzia, Use of deep learning for automatic low-cost detection of cracks in tunnels, 2018.*
2) *Master's thesis: Luca Zacheo, Deep Learning for automatic crack detection inside tunnels - Second Prototype, 2019.*
3) *Master's thesis: Antonio Marangi, Simultaneous Localization and Mapping, 2019.*
4) *Master's thesis: Nicola Bruno, User Interface for Autonomous Driving Robot, 2019.*
5) *Master's thesis: Fausto Lizzio, Mechanical Design of an Autonomous Driving Robot, 2019.*

Workshops

NA

Software

NA

4 Outcomes

1) *We have designed, developed and build two low cost data acquisition platforms, one video based and the other image based for taking video or images of the interior lining of tunnels. Both have been successfully field tested in several tunnels in the Los Angeles area. The second prototype has been build with a tiltable platform that allows for changing the orientation of the acquisition system relative to the tunnel surface. While outside the scope of the current work, it is not difficult to motorize the platform and develop a feedback system for alignment relative to the surface being imaged.*
2) *We have designed, developed and build a GPU based vSLAM system for localization indoors in the absence of any GPS or beacon signals of any kind. The unit has been successfully tested both in the Engineering building of CSLA for assessing localization accuracy and in the field at Griffith park's tunnel.*
3) *We have successfully trained two Neural Network systems and verified the system's ability to automatically and autonomously detect cracks from the database of images produced. One is a pre-trained deep CNN by Google (Inception v4), the other is custom build in-house system with all layers trained by CSLA team.*

4 Impacts

1) *From project inception to end of 2019, five students participated in the project and have completed their Master's thesis on the project topic or a related area. Two are pursuing Ph.D. degrees and the others have been successfully employed. From Fall 2019 to present we have involved five additional students in the project.*
2) *Department of Electrical and Computer Engineering (ECE) at CSLA is developing a second course on Machine Learning and one of the key topics is projected to be deep Convolutional Neural Networks (CNN). Application examples of the technology in different areas will be an important element of the course. We plan to highlight and exemplify use of deep CNNs in infrastructure monitoring.*

APPENDIX B - DATA FROM THE PROJECT

Following data was generated as a result of the R&D conducted:

1) With the first data acquisition trial using video based prototype-1, it has been possible to collect videos from inside of tunnels in Los Angeles lasting approximately 40 minutes (total time: 40.09, total number of frames: 2405.4 x 30 = 72162 frames). After an accurate selection, it has been possible to obtain a dataset of 6494 images.
    a. Crack detection folder: 4094
    b. No-Cracks detection folder: 2400
    c. Total dataset images: 6494
2) Prototype-2 data acquisition system was based on DSLR camera. Using this prototype, we managed to obtain a high resolution dataset composed of 1024 pictures of the interior wall of tunnels in Los Angeles, of which 521 show surfaces with potential cracks. This image dataset were artificially expanded and used to train Inception-v4 deep CNN.
3) Results of tests conducted on Inception-v4 deep CNN using different hyper parameters using video based data are reported below:

Test Number	Iterations	Learning Rate	Train Batch Size	Cross Entropy	Test Accuracy
1	4000	0.01	100	0.143	95%
2	9000	0.01	100	0.109	97.5%
3	9000	0.001	100	0.132	96.4%
4	15000	0.001	100	0.147	95.1%
5	15000	0.01	100	0.103	97.6%
6	10000	0.01	100	N.A.	N.A.
7	9000	0.01	300	0.102	97.6%
8	9000	0.01	1000	0.106	97.5%
9	9000	0.01	1000	0.094	98.1%

4) Optimized parameters obtained through extensive testing for the custom Artificial Neural Network trained using video based data to detect cracks on surface of tunnel walls for different layers of the network are tabulated below:

Section	Input	Output	Patch	Strike	Filters
Input Layer	1920x1080x3	128x128	N.A.	N.A.	3
1th CL	128x128x3	128x128	5x5	1	32
1th PL	128x128x32	64x64	2x2	1	32
2nd CL	64x64x32	64x64	5x5	1	64
2nd PL	64x646x64	32x32	2x2	1	64
FCL	64x64x32	1024	N.A.	N.A.	N.A.
Softmax Layer	1024	2	N.A.	N.A.	N.A.

5) Results of tests conducted on Inception-v4 deep CNN used for image classification using data generated by the second prototype are tabulated below:

Test Number	**Iterations**	**Learning Factor**	**Train Batch Size**	**Validation Batch Size**	**Evaluation Interval**	**Cross Entropy**	**Test Accuracy**
1	4500	0.1	100	202	100	0.002974	92.7%
2	2000	0.1	100	100	100	0.002808	93.6%
3	1000	0.1	100	100	100	0.009057	91.7%
4	2500	0.01	100	100	100	0.02428	93.6%
5	6000	0.001	100	100	100	0.07742	94.5%
6	9000	0.001	100	100	100	0.104863	93.6%
7	3000	0.1	50	100	100	0.002662	91.7%
8	2000	0.1	75	100	100	0.003804	92.7%
9	2000	0.1	150	100	100	0.006381	92.7%
10	9000	0.001	50	100	100	0.055273	93.6%

6) Relative localization error parameters associated with vSLAM based on experiments conducted in the Engineering building of CSLA is tabulated below:

Seq.1:		$\mu_{relerrx}$	$\mu_{relerrz}$	μ_{errx}	μ_{errz}	σ_x^2	σ_z^2	L
• 982 Frames	Trial1	2.02%	1.53%	0.6 mm	1.6 mm	1.3 E-4	1.9 E-4	0
• Map. in 4.5 min	Trial2	2.21%	1.17%	0.7 mm	1.7 mm	1.2 E-4	2 E-4	0
• 30 FPS	Trial3	2.49%	0.83%	0.9 mm	2.1 mm	1.4 E-4	1.9 E-4	0
Seq.2:		$\mu_{relerrx}$	$\mu_{relerrz}$	μ_{errx}	μ_{errz}	σ_x^2	σ_z^2	L
• 1039 Frames	Trial1	0.67%	1.56%	0.8 mm	2.2 mm	8 E-5	1.7 E-4	0
• Map. in 5.1 min	Trial2	2.32%	2.11%	1.2 mm	1.3 mm	9 E-5	1.7 E-4	0
• 30 FPS	Trial3	2.27%	1.76%	1.6 mm	0.13 mm	7.8 E-5	1.5 E-4	0