MULUNGUSHI UNIVERSITY

**Pursuing the frontiers of Knowledge**

SCHOOL OF ENGINEERING & TECHNOLOGY

Department OF Computer Science & Information Technology

Project Report Title**: AUTOMATED ROAD DAMAGE DETECTION AND CLASSIFICATION SYSTEM USING CONVOLUTIONAL NEURAL NETWORKS**

Name**: Chanda M. Chimfwembe**

Student No: **202001198**

Course: ICT431 Capstone Project (Presentation Stage)

Programme: Bachelor of Computer Science

Supervisor: **Dr. B. Halubanza**

**REPORT SURRENDERED IN PARTIAL FULFILLMENT OF THE NECESSITIES NEEDED FOR THE AWARD OF BACHELOR OF SCIENCE DEGREE IN COMPUTER SCIENCE FOR THE 2023/2024 ACADEMIC YEAR**

# INTRODUCTION

## Background and Summary

Road infrastructure plays a pivotal role in facilitating transportation, enabling the seamless movement of people and goods (Haghani et al., 2022). However, roads are susceptible to environmental and usage-related factors, including severe weather conditions, heavy traffic, and natural wear and tear, all contributing to road deterioration (Huang et al., 2023). Diagnosing road deterioration is crucial for sustaining the economic vitality of Zambia, ensuring the well-being of the local population, and advancing secure and adequate transportation.

Traditionally, visual inspections conducted by trained personnel over extended periods have been the primary method for assessing road deterioration. However, this approach is time-consuming, expensive, and subjective, depending on the expertise of the inspector (Zhang et al., 2021). In response to these challenges, automated techniques for identifying road damage, leveraging technologies such as remote sensing, computer vision, and machine learning, have gained attention. These technologies provide an opportunity to revolutionize road maintenance and repair in Zambia (Zhang et al., 2022).

The development of an Automated Road Damage Detection and Classification System using Convolutional Neural Networks specifically tailored for Zambia addresses the country's unique challenges. Tailoring the system to local conditions, such as climate, road materials, and traffic patterns, ensures its effectiveness in Zambia's diverse environment. This customization enhances the system's adaptability and relevance to Zambia's road infrastructure.

### Challenges in Zambian Road Infrastructure

Zambia's road network, spanning diverse landscapes and facing varied environmental conditions, presents unique challenges for maintenance and repair. The extensive road networks, coupled with limited resources, make it imperative to explore innovative and cost-effective solutions. Conventional methods, such as manual inspections, are resource-intensive, time-consuming, and subject to interpretation bias. Zambia needs a modern and efficient approach to address the maintenance demands of its road infrastructure.

The economic implications of road damage in Zambia are significant. Timely and accurate detection of road deterioration is crucial for preserving the economic vitality of regions dependent on well-maintained roads. Efficient transportation is fundamental for commerce and trade, and any disruptions due to road deterioration can result in increased costs and delays. An automated system that swiftly identifies and classifies road damage can lead to proactive and cost-effective maintenance, reducing economic losses associated with road disruptions.

The well-being of the local population is directly linked to road safety. Damaged roads can pose hazards to drivers, passengers, and pedestrians. Rapid identification of road damage allows for prompt repairs, mitigating safety risks and ensuring secure transportation. This is particularly crucial in Zambia, where road networks connect various communities, and any disruptions can impact the accessibility of essential services.

Efficient road maintenance contributes to environmental sustainability. Timely repairs prevent the escalation of road damage, reducing the environmental footprint associated with extensive repairs. Additionally, early detection of issues like potholes helps minimize soil erosion and water contamination, contributing to overall environmental conservation.

### Types of Damages of Roads in the Zambian Context

The classification of road damage is crucial for prioritizing maintenance and repairs in Zambia. Various damages, such as cracks, rutting, roughness, and potholes, have unique implications for the environment and vehicular traffic [[2-6]](#_REFERENCES).

1. **Cracks**: Minor surface damage that may grow into larger holes if not repaired promptly, with causes including poor-quality materials and workmanship, inadequate pavement thickness, subsidence, and pavement age.
2. **Rutting**: Depressions or grooves in the road surface caused by large loads, repetitive driving, and poor drainage.
3. **Roughness**: Irregular road surfaces uncomfortable for drivers and passengers, caused by factors like lack of aggregate purity, poor adhesion of surface treatment to the base, inadequate aggregate penetration, subpar premix quality, and inconsistent binder distribution.
4. **Potholes**: Small to medium-sized holes caused by traffic wear and tear, weather conditions, and inadequate maintenance.

## Problem Statement

Despite the critical role that road infrastructure plays in Zambia's economic development and community well-being, the current methods of road damage detection and classification pose significant challenges. The manual inspection processes, traditionally employed, are resource-intensive, time-consuming, and subject to interpretation bias. Zambia's extensive road networks, coupled with limited resources, accentuate the urgency for a more efficient and technologically advanced solution.

* **Resource Limitations:** Limited resources hinder the effectiveness of manual inspection methods, making it challenging to cover the vast road networks adequately. This limitation results in delays in identifying and addressing road damage promptly.
* **Subjectivity and Interpretation Bias:** Manual inspections are inherently subjective, relying on the expertise and interpretation of inspectors. This subjectivity introduces a level of inconsistency in identifying and classifying road damage, impacting the reliability of the assessments.
* **Economic Impact:** Undetected road damage poses significant economic implications. Delays in identifying deterioration led to increased repair costs, impacting both regional economic vitality and transportation efficiency critical for trade and commerce.
* **Safety Concerns:** Damaged roads pose safety risks to drivers, passengers, and pedestrians. The current challenges in promptly identifying road damage jeopardize road safety and the accessibility of essential services connected by Zambia's road networks.
* **Environmental Impact:** Inefficient detection and delayed repairs contribute to a higher environmental footprint associated with extensive road repairs. Early identification of road issues is crucial for minimizing soil erosion, water contamination, and overall environmental conservation.

Addressing these challenges requires a shift towards a more advanced and automated road damage detection and classification system. Leveraging technologies like Convolutional Neural Networks (CNNs) tailored to Zambia's unique conditions holds the key to overcoming the current limitations. This solution aims to streamline the detection process, enhance accuracy, and optimize resource allocation, ultimately contributing to the efficient and sustainable management of Zambia's road infrastructure.

## Aim

The primary aim of this study is to design, develop, and implement an Automated Road Damage Detection and Classification System tailored for Zambia, utilizing Convolutional Neural Networks (CNNs). This system aims to address the inherent challenges in the current manual inspection methods and introduce a more efficient, accurate, and technology-driven approach to road infrastructure management.

## Objective

The goals of the suggested project are as follows:

1. To identify the challenges faced by road damages
2. Gather and clean the dataset.
3. To train the model, prepose and select the ideal model, and validate the chosen model
4. To develop and integrate the model with a mobile app.

## Summary/ Conclusion

This chapter makes a commencement of the project titled “Real-Time Detection and Classification of Road Damage using Deep Learning”. It offers a glimpse into the project’s core aim, objectives, and rationale. This summary succinctly captures the project’s scope, it’s underlying purpose and delineates five distinct objectives. Centered on the deployment of deep learning techniques for the real-time identification and categorization of road damage, this chapter lays the foundation for a thorough examination of this pioneering methodology in the realm of road infrastructure management.

# LITERATURE REVIEW

## Introduction

This chapter is dedicated to the evaluation and analysis of current models in Real-Time Detection and Classification of Road Damage using Deep Learning. A detailed examination of the operational mechanisms of existing systems will be conducted, with a focus on drawing insightful comparisons with the proposed system. The primary objective of this chapter is to provide a comprehensive summary of the literature review, shedding light on the acquired insights and the pertinence of existing models to the proposed system. Through this thorough analysis, a nuanced understanding of the road damage detection landscape will be established, facilitating the development and refinement of the proposed system.

## Review of existing/current system

The real-time detection and classification of road damage hold paramount importance for effective decision-making in road infrastructure management. A robust model for accurate road damage prediction is crucial for swift and informed decision-making by authorities. Diverse methodologies exist for predicting road damage, and this review article investigates the landscape of deep learning applications in the real-time detection and classification of road damage within the existing literature.

### Road damage detection using image processing

Several attempts have been made to develop a method for analyzing road properties by using a combination of recordings by in-vehicle cameras and image processing technology to more efficiently inspect a road surface. For example, a previous study proposed an automated asphalt pavement crack detection method using image processing techniques and a naive Bayes based machine-learning approach (Chun et al., 2015).

In addition, a pothole detection system using a commercial black-box camera has been previously proposed (Jo and Ryu, 2015). In recent times, it has become possible to quite accurately analyze the damage to road surfaces using deep neural networks (Zhang et al., 2016; Maeda et al., 2016; Zhang et al., 2017; Fan et al., 2018). For instance, Zhang et al. (2017) introduced Crack Net, which predicts class scores for all pixels. However, such road damage detection methods focus only on the determination of the existence of damage. Though some studies do classify the damage based on types—for example, Zalama et al. (2014) classified damage types vertically and horizontally, and Akarsu et al. (2016) categorized damage into three types, namely, vertical, horizontal, and crocodile—most studies primarily focus on classifying damages between a few types. There are other studies that detect blurry road markings (Kawano et al., 2017), and classify the cracks and sealed cracks (Zhang et al., 2018). Therefore, for a practical damage detection model for use by municipalities, it is necessary to clearly distinguish and detect different types of road damage; this is because, depending on the type of damage, the road administrator needs to follow different approaches to rectify the damage.

Furthermore, the application of deep learning for road surface damage identification has been proposed by a few studies, for example, studies by Maeda et al. (2016) and Zhang et al. (2016). However, the method proposed by Maeda et al. (2016), which uses 256 ×256 pixel images, identifies the damaged road surfaces,

but does not classify them into different types. In addition, the method of Zhang et al. (2016) identifies whether damage occurred exclusively using a 99 × 99 patch obtained from a 3,264 × 2,448-pixel image. Further, a 256 × 256-pixel damage classifier is applied using a sliding window approach (Felzenszwalb et al., 2010) for 5,888 × 3,584-pixel images to detect cracks on the concrete surface (Cha et al., 2017). In these studies, classification methods are applied to input images and damage is detected. Recently, it has been reported that object detection using end-to-end deep learning is more Road damage detection and classification 3 accurate and has a faster processing speed than using a combination of classification methods. As an example of a method using end-to-end deep learning performing better than tradition methods, white line detection based on end-to-end deep learning using OverFeat (Sermanet et al., 2013) outperformed a previously proposed empirical method (Huval et al., 2015). However, to the best of our knowledge, no example of the application of end-to end deep learning method for road damage detection exists. It is important to note that classification refers to labeling an image rather than an object, whereas detection means assigning an image a label and identifying the object’s coordinates as exemplified by the ImageNet competition (Deng et al., 2009). The term “end-to-end” indicates that input and output relationships are trained directly with a single model.

### Road damage detection using smartphones

In general, vehicles designed specifically for road inspection are expensive. Meanwhile, mobile devices such as smartphones have made remarkable progress in recent years, and examples of road inspection using smartphone sensors are increasingly common. Using a smartphone is advantageous insofar as it is possible to inspect the road surface cheaply and exhaustively. For example, Buttlar and Islam (2014) proposed a method to measure the flatness of a road using the accelerometer of a smartphone installed in a car. Furthermore, Casas-Avellaneda and L´opez-Parra (2016) proposed a method that visualizes (on a map) potholes detected by smartphone sensors. In addition, Mertz et al. (2014) proposed a method to handle road images acquired by on-board smartphones installed on cars that operate on a daily basis, such as general passenger automobiles, buses, and garbage trucks, to detect road surface damage with an external laptop.

To the best of our knowledge, however, there is no research on processing road images acquired by smartphones to detect road damage. Therefore, we demonstrate that using end-to-end deep learning is feasible for processing such images.

### Image data set of road surface damage

Although an image data set of the road surface exists, called the KITTI data set (Geiger et al., 2013), it is primarily used for applications related to automated driving. There is also the GAP data set for road damage detection with features of around 2,000 high-resolution images with manually annotated damage (six classes) (Eisenbach et al., 2017). The GAP data set is the only publicly available data set for road damage detection. In all the studies focusing on road damage detection, the researchers independently proposed unique methods using acquired road images. Therefore, a comparison between the methods presented in these studies is difficult.

Furthermore, according to Mohan and Poobal (2017), there are few studies that construct damage detection models using real data, and 20 of these studies use road images taken directly from above the road. For instance, the images of the GAP data set were taken from above the road. In fact, it is difficult to reproduce the road images taken from directly above, because doing so involves installing a camera outside the car body, which, in many countries, is a violation of the law; in addition, it is costly to maintain a dedicated car solely for road images.

### Object detection system

In a CNN-based object detection method such as R-CNN (Region-based Convolutional Neural Networks;

Girshick et al., 2014) and Fast R-CNN (Girshick, 2015), it is necessary to obtain the object candidate region

in advance by using another method, such as selective search (Uijlings et al., 2013) or BING (Binarized Normed Gradients; Cheng et al., 2014). For this reason, the process is slow and the judgment accuracy is relatively low, insofar as it is a two-stage process (i.e., the candidate area is detected and the detected area is classified by a CNN). On the other hand, Faster R-CNN (Ren et al., 2015) makes it possible to train the model end-to-end, and the accuracy of determination and the execution speed can be improved by using the Region Proposal Network, which performs object candidate region detection. Furthermore, rather than cropping features from the same layer where the region proposals are predicted—as in the case of the Faster R-CNN method—the R-FCN (Region based Fully Convolutional Networks) method proposed by Dai et al. (2016) crops from the last layer of features prior to prediction. This approach of pushing cropping to the last layer minimizes the amount of per region computation that must be performed. Dai et al. (2016) showed that their R-FCN model (using Resnet 101) could achieve accuracy comparable to Faster R-CNN, and often at faster running speeds. Although the processing speed has been greatly improved by the above method, the computational load is somewhat large when processing images from modern mobile devices.

YOLO (You Only Look Once) (Redmon et al.,2016; Redmon and Farhadi, 2017) is an object detection framework that can achieve high mean average precision (mAP) and speed. In addition, YOLO can predict

the region and class of objects with a single CNN. An advantageous feature of YOLO is that its processing

speed is considerably fast, because it solves the problem as a mere regression, detecting objects by considering background information. The YOLO algorithm outputs the coordinates of the bounding box of the object candidate and the confidence of the inference after receiving an image as input. Furthermore, SSD (Single Shot MultiBox Detector; Liu et al., 2016) is an object detection framework that uses a single feed-forward convolutional network to predict classes directly and anchor offsets without requiring a second stage per proposal classification operation. The key feature of this framework is the use of multiscale convolutional bounding box outputs attached to multiple feature maps at the top of the network. With this key feature, SSD is fast and has fewer errors than YOLO. In this research, SSD is adopted as a training algorithm for processing images from a mobile terminal.

## Comparisons of Reviewed Systems

This section will focus on comparing the three systems reviewed and compare some of the specification and functionalities with proposed system. The table below shows a table containing the three systems in question and the proposed system.

Table 2‑1 Comparison of systems with Proposed System

|  |  |  |  |
| --- | --- | --- | --- |
| **System (Research Paper)** | **Road damage detection using image processing** | **Road damage detection using smartphones** | **Proposed System** |
| **System Task** | Predicting | Predicting | Predicting & Classifications |
| **Implementation Language(s)** | Python | Not Specified | Python & Java (Mobile App) |
| **Model Type** | Deep Neural Networks | Machine Learning | Deep Neural Network |
| **Dataset** | Road photos | Road photos | Road surface images |
| **Application Type** | Web App | Mobile App | Mobile App |
| **Data Requirement** | High | Medium | Medium |
| **Available in Zambia** | NO | NO | YES |
| **Real Time** | NO | YES | YES |

# RESEARCH METHODOLOGY

## Introduction

A software development process is said to be a set of activities, methods, practices and transformations that software engineers and users use to develop and maintain software products (Tanrıöver & Demirörs, 2015). There are four fundamental activities that are involved and common to all software processes. These activities are:

* Software specification phase; where the customers and engineers define the software that is to be formed and the constraints on its function.
* Software development phase; where the software is designed and programmed.
* Software validation phase; where the software is checked to ensure that it is what the client requires.
* Software evolution phase; where the software is customized to suit the new requirements of the users.

A number of software development life cycle (SDLC) models have been created: waterfall,

spiral, V-Model, rapid prototyping, incremental, and synchronize and stabilize. The proposed

system will use the waterfall methodology

## Selected Methodology (with step by step)

### Waterfall Model

This is the process that will be utilized to construct the proposed system. The initial software development process model was drawn from more broad system engineering methods. The 'waterfall model' or software life cycle is named after the way the phases flow from one to the next. In essence, the waterfall model is a plan-driven process since it requires you to plan and schedule all of the process tasks before you begin working on them. (Sommerville, 2011).

Development of Road Damage Detection and Classification System.

1. **Requirements analysis and definition,** at this stage we gathered all the requirements needed to build the system and they have been well documented. All the constraints, services and goals are created with the consultation of the system users.
2. **System and software design**, since this is only a software application and no hardware is needed, this means that during the system design process we shall only allocate the requirements to the software system by establishing an overall system architecture. At this stage we shall strike a balance between requirements that conflict with each other for example maintainability and performance, within our implementation environment.
3. **Implementation and unit testing**, at this stage our completed design will be translated into program code but inform of a set of programs or program units. During this stage we shall use anaconda and the python language for training and testing the machine learning classifier. Then we shall test each unit to see if it meets the specification. At this stage we shall test the mobile app’s functionalities.
4. **Integration and system testing**, at this stage the individual program units that we tested for example our classifier will be integrated into one unit and tested as complete system, at this stage the Classifier and the mobile app will be tested as a single unit. After a successful testing it shall be delivered to the customer.
5. **Operation and maintenance**, at this stage the system will be put through practical use, and then it shall undergo maintenance which will involve correcting errors which will be discovered that could not be seen earlier on, then improving the system.

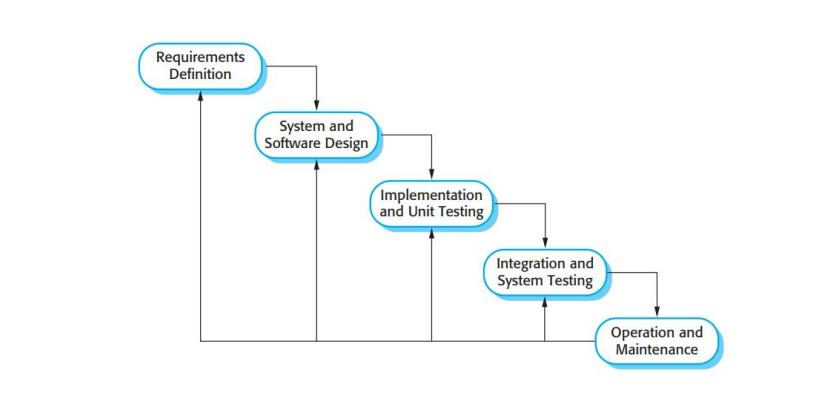


Figure 3.1 Waterfall model (Sommerville, 2011)

## Justification of Selected Methodology

The Waterfall Model is the most basic SDLC approach for software development. A sequential model is exemplified by this model. As a result, it's also known as a linear sequential life cycle model (GeeksforGeeks, 2020). The waterfall model is picked as the main methodology because it suits this proposed system very well in that, the proposed system is well documented and will have each phase clearly defined before the building of the system even begins.

Our system development process has two parts which is research on the classifiers and building the mobile app, since the waterfall model is done in phases and makes sure each phase is completed before going to the next phase it suits this system because the evaluating, Pre-processing, training and testing of our classifier has to be done in steps and each phase must be completed fully for us to go to the next phase, then after the research phase is completed fully we then have to build the mobile app as our next phase in order for it to work with the classifier. Building the mobile app will also be done in phases as well as well as having all the requirements fully documented and as clear as possible.

The tools that will be used are well understood, this point works well under the waterfall model. Since the project is a 2-semester long project it makes it short as compared to most software projects, the proposed system won’t be expensive as well which also fits the waterfall model methodology.

# PROJECT MANAGEMNT

This chapter delineates the essential components of budgeting and scheduling uniquely designed for the envisioned Real-Time Detection and Classification of Road Damage using Convolutional Neural Networks. In the realm of project management, the term "risk" traditionally pertains to factors that may hinder the realization of project objectives (Schwalbe, 2015). However, for the scope of this chapter, the focus will be on budgeting and scheduling considerations, with a deliberate exclusion of risk analysis.

Effort costing in the development of a road damage detection system entails predicting the most realistic amount of effort, expressed in person-hours, required for the software's creation or maintenance. This prediction process relies on dealing with incomplete, ambiguous, and noisy data (Schwalbe, 2015).

Within the confines of this chapter, the spotlight will be on budgeting and scheduling tailored for the proposed Road Damage Detection System. It will elucidate the pivotal role of effective budget management and scheduling in achieving project success. The chapter will introduce a comprehensive budget plan, delineating financial considerations and resource allocation specifics. Scheduling aspects will be emphasized, providing a well-structured timeline for pivotal project milestones and activities. Additionally, the chapter will delve into methodologies for estimating budgetary requirements and navigating scheduling constraints inherent in the development of the Road Damage Detection System.

## BUDGET CALCULATION

The budget outline for the project is shown in the table below:

Table 4‑1 Project Budget

|  |  |
| --- | --- |
| NAME | Amount (ZMK) |
| Salary | 5000/month |
| Computer (with required software) | 12000 |
| Power | 60/month |
| Internet | 450/month |
| Other Costs | 1000 |

Calculating budget total:

Total wage for the project is (K5000 for 8 months) / 5 = ***K 8,000***

The total budget of the project is

8,000 + 12,000 + (60 ∗ 8) + (450 ∗ 8) + 1000 = ***𝐾 25,080***

The project duration is 8 months which we break down to 4 months in each semester. The

project is further broken down using the table below showing each activity and expenditure.

Table 4‑2 Waterfall model-based Budget

|  |  |
| --- | --- |
| Activity | Amount (ZMK) |
| Feasibility | 1000 |
| Requirement analysis and Gathering | 1000 |
| Design | 2500 |
| Coding | 2000 |
| Testing | 1000 |
| Report Writing and Presentation | 500 |
| Computer (with required software) | 12000 |
| Power | 480 |
| Internet | 3600 |
| Other Costs | 1000 |
| Grand Total | 25,080 |

## Scheduling and Work plan

Scheduling is the process of organizing, controlling and optimizing work and workloads in a production or manufacturing process. Scheduling is used to allocate resources and processes.

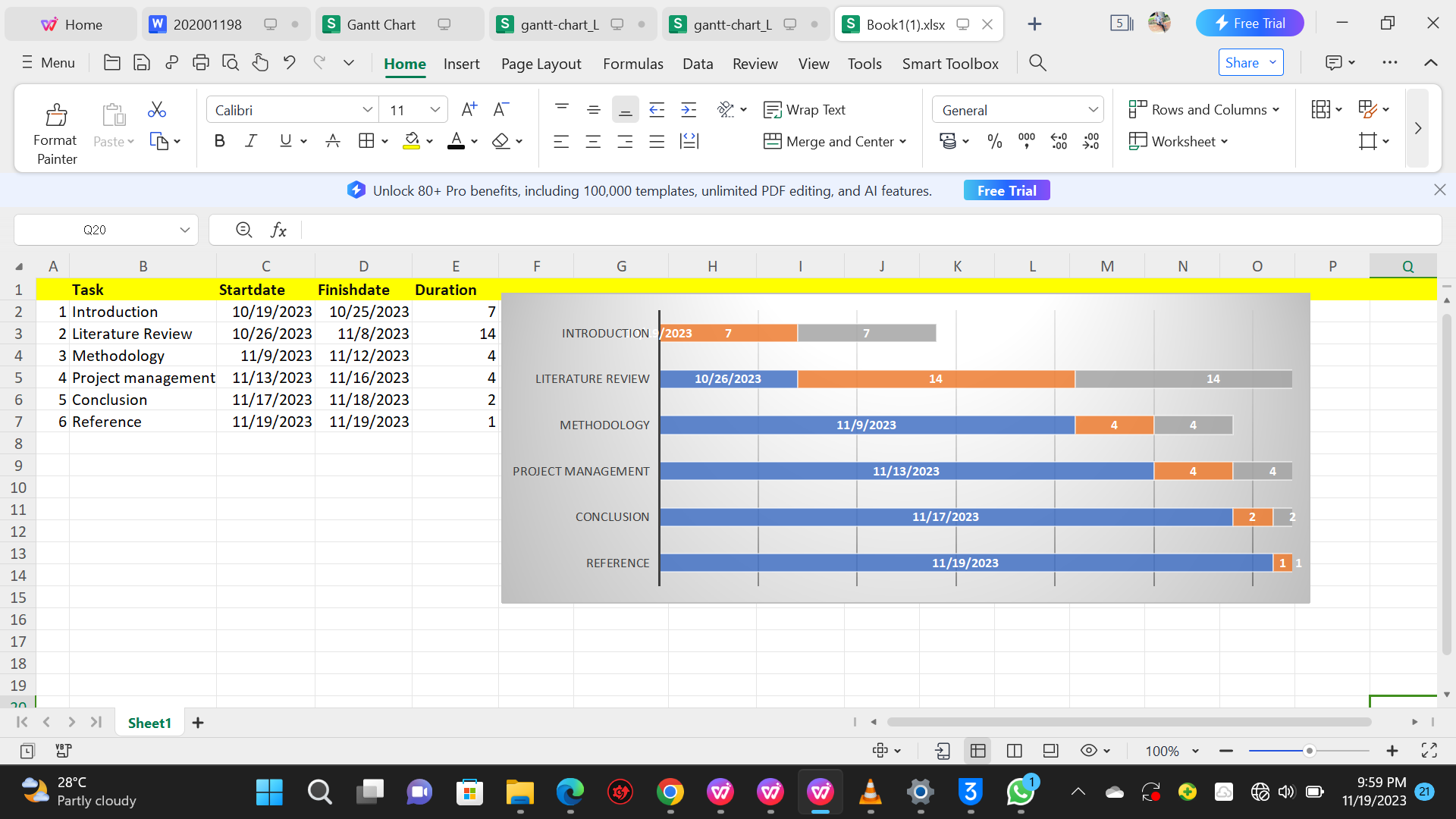


Figure 4.1 The Proposal Gantt Chart

## Summary

This chapter delves into the critical components of risk analysis and effort costing, integral for assessing potential project risks and estimating the necessary resources. The focus on risk analysis is to identify and proactively mitigate challenges that may arise during the project. Simultaneously, effort costing is utilized to determine the time and resources required for the timely completion of the Road Damage Detection and Classification system.

The incorporation of the COCOMO model enhances the precision of estimating software costs, offering valuable insights for effective budgeting and resource allocation. Furthermore, the use of a Gantt chart provides a visual roadmap, aiding in project timeline planning and detailing key development stages at specific intervals. Together, these tools contribute to efficient project management, ensuring a structured approach for the successful delivery of the Road Damage Detection and Classification project.

# CONCLUSION

Chapter 5 provides a thorough exploration of the pivotal aspects of risk analysis and effort costing, essential components for effective project planning in the Real-Time Detection and Classification of Road Damage. The chapter leverages the COCOMO model to estimate software costs, offering valuable insights into budgeting and resource allocation. Additionally, a Gantt chart is employed to meticulously outline the project timeline, delineating specific development stages at predefined intervals.

In summary, this chapter serves as a comprehensive guide within the project proposal for the development of a Real-Time Road Damage Detection and Classification System using Convolutional Neural Networks. It encompasses the project's significance, the associated challenges, a review of existing methods, the proposed system's architecture, the chosen software development methodology, project management specifics, and the crucial considerations of risk analysis and effort costing.

## Research Contributions

The envisioned Road Damage Detection and Classification System make substantial contributions to the field of infrastructure maintenance. It empowers non-specialist users, including road maintenance personnel, to efficiently detect and classify road damages in real-time. This system facilitates informed decision-making in matters of road management, resource allocation, and financial planning for maintenance activities. Its impact is particularly significant in regions where access to road infrastructure specialists may be limited, showcasing widespread applicability through the integration of Convolutional Neural Networks for precise and swift detection of road damages.

## References

1. Akarsu, B., Karakose, M., Parlak, K., Erhan, A. K. I. N. & Sarimaden, A. (2016), A fast and adaptive road defect detection approach using computer vision with real-time implementation, International Journal of Applied Mathematics, Electronics and Computers, 4(Special Issue-1), 290–95.
2. Anon, 1994. International road maintenance handbook: practical guidelines for rural road maintenance. 3. Paved roads[M]. Transport Research Laboratory, UK.
3. Arya D, Maeda H, Ghosh S K, et al., 2020. Global road damage detection: State-of-the-art solutions. 2020 IEEE International Conference on Big Data (Big Data). 5533-5539.
4. Arya D, Maeda H, Ghosh S K, Toshniwal D, Mraz A, et al., 2021. Deep learning-based road damage detection and classification for multiple countries. Automation in Construction, 132:103935.
5. Arya D, Maeda H, Ghosh S K, Toshniwal D, Sekimoto Y, 2021. RDD2020: An annotated image dataset for automatic road damage detection using deep learning. Data in brief, 36:107133.
6. Arya D, Maeda H, Ghosh S K, et al., 2022. RDD2022: A multi-national image dataset for automatic Road Damage Detection[J]. arXiv preprint arXiv:2209. 08538.
7. Bannerman, P.L. (2008) ‘Risk and risk management in software projects: A reassessment’, Journal of systems and software, 81(12), pp. 2118–2133.
8. Boehm, B. et al. (1995) ‘Cost models for future software life cycle processes: COCOMO 2.0’, Annals of software engineering, 1(1), pp. 57–94.
9. Casas-Avellaneda and L´opez-Parra (2016) proposed a method that visualizes (on a map) potholes detected by smartphone sensors.
10. Cha, Y.-J., Choi, W. & Buyukozturk, O. (2017), Deep learning-based crack damage detection using convolutional neural networks, Computer-Aided Civil and Infrastructure Engineering, 32(5), 361–78.
11. Chun, P.-J., Hashimoto, K., Kataoka, N., Kuramoto, N. & Ohga, M. (2015), Asphalt pavement crack detection using image processing and naive Bayes based machine learning approach, Journal of Japan Society of Civil Engineers, Ser. E1 (Pavement Engineering), 70(3), 1–8.
12. Dhaiphule, Sakshi. (2023). Review of AI based Techniques for Road Damage Detection. International Journal for Research in Applied Science and Engineering Technology. 11. 4776-4786. 10.22214/ijraset.2023.54532.
13. Eisenbach, M., Strick Buttlar, W. G. & Islam, M. S. (2014), Integration of Smart Phone-Based Pavement Roughness Data Collection Tool with Asset Management System, Technical Report, US DOT Region V Regional University Transportation Centre, NEXTRANS Centre, West Lafayette, IN.
14. Fan, Z., Wu, Y., Lu, J. & Li, W. (2018), Automatic pavement crack detection based on structured prediction with the convolutional neural network, Computer Vision and Pattern Recognition, arXiv preprint arXiv:1802.02208.
15. Felzenszwalb, P. F., Girshick, R. B., McAllester, D. & Ramanan, D. (2010), Object detection with discriminatively trained part-based models, IEEE Transactions on Pattern Analysis and Machine Intelligence, 32(9), 1627–45.
16. GeeksforGeeks. (2020, August 5). Waterfall Model in Software Development. Retrieved from https://www.geeksforgeeks.org/waterfall-model/Geiger, A., Lenz, P., Stiller, C. & Urtasun, R. (2013), Vision meets robotics: the KITTI dataset, The International Journal of Robotics Research, 32(11), 1231–37.
17. Hatmoko J, Setiadji B, Wibowo M, 01 2019. Investigating causal factors of road damage: a case study[J]. MATEC Web of Conferences, 258: 02007.
18. Haghani M, Behnood A, Dixit V, et al., 2022. Road safety research in the context of low and middle-income countries: Macro-scale literature analyses, trends, knowledge gaps and challenges. Safety science, 146(105513): 105513.
19. Hatmoko J, Setiadji B, Wibowo M, 01 2019. Investigating causal factors of road damage: a case study[J]. MATEC Web of Conferences, 258: 02007.
20. Huang Y, Wu J, Luo J. Pavement condition assessment using machine learning techniques based on big data: A review. Journal of Transport Engineering, Part B: Pavements, 2023;150(2): 04021156.
21. Kawano, M., Mikami, K., Yokoyama, S., Yonezawa, T. & Nakazawa, J. (2017), Road marking blur detection with drive recorder, in Proceedings of the 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, 4092–97.
22. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y. & Berg, A. C. (2016), SSD: single shot multi-box detector, in Proceedings of the European Conference on Computer Vision, Springer, Berlin, 21–37.
23. Lux, T. C. H., Nagy, S., Almanaa, M., Yao, S. & Bixler, R. (2019), "A Case Study on a Sustainable Framework for Ethically Aware Predictive Modelling," 2019 IEEE International Symposium on Technology and Society (ISTAS), Medford, MA, USA, 2019, pp. 1-7, doi:10.1109/ISTAS48451.2019.8937885.
24. Maeda, H., Sekimoto, Y. & Seto, T. (2016), Lightweight Road manager: smartphone-based automatic determination of road damage status by deep neural network, in Proceedings of the 5th ACM SIGSPATIAL International Workshop on Mobile Geographic Information Systems, Burlingame, CA, 37–45.
25. Mertz, C., Varadharajan, S., Jose, S., Sharma, K., Wander, L. & Wang, J. (2014), City-wide Road distress monitoring with smartphones, in Proceedings of ITS World Congress, Tokyo, Japan, 1–9.
26. Mohan, A. & Poobal, S. (2017), Crack detection using image processing: a critical review and analysis, Alexandria Engineering Journal, <https://doi.org/10.1016/j.aej.2017.01.020>.
27. Pregnolato M, Ford A, Wilkinson S M, et al., 2017. The impact of flooding on road transport: A depth-disruption function[J]. Transportation Research Part D: Transport and Environment, 55: 67-81.
28. Redmon, J. & Farhadi, A. (2017), YOLO9000: better, faster, stronger, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI.
29. Redmon, J., Divvala, S., Girshick, R. & Farhadi, A. (2016), You only look once: unified, real-time object detection, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Nevada, 779–88.
30. Ren, S., He, K., Girshick, R. & Sun, J. (2015), Faster RCNN: towards real-time object detection with region proposal networks, in Proceedings of the Advances in Neural Information Processing Systems, Montreal, Canada, 91–99.
31. Royce, W. W. (1970). Managing the development of large software systems: Why it is late. In Proceedings of the 9th ACM conference on Data Storage Retrieval (pp. 328-338).
32. Schwalbe, K. (2015) Information technology project management. Cengage Learning.
33. Schwalbe, K., 2013. Information Technology Project Management, revised (with Premium Online Content Printed Access Card).
34. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R. & LeCun, Y. (2013), OverFeat: integrated recognition, localization and detection using convolutional networks, Computer Vision and Pattern Recognition, arXiv preprint arXiv:1312.6229.
35. Sommerville, I., 2011. Software engineering. 9th ed. s.l.: Addison-Wesley /Pearson.
36. Tanrıöver, P., & Demirörs, O. (2015). Software Development Methodologies: Concepts, Techniques, and Tools. IGI Global.
37. TechTarget. (2022, March 29). Waterfall Model. Retrieved from <https://www.techtarget.com/searchsoftwarequality/definition/waterfall-model/>
38. Trendowicz, A. and Jeffery, R. (2014) ‘Software project effort estimation’, Foundations and Best Practice Web Appdelines for Success, Constructive Cost Model–COCOMO pages, 12, pp. 277–293.
39. Tutorialspoint. (2022, July 12). Waterfall Model in SDLC. Retrieved from <https://www.tutorialspoint.com/adaptive_software_development/sdlc_waterfall_model.html>
40. Uijlings, J. R., Van De Sande, K. E., Gevers, T. & Smeulders, A. W. (2013), Selective search for object recognition, International Journal of Computer Vision, 104(2), 154–71.
41. Van Wyngaard, C.J., Pretorius, J.-H.C., and Pretorius, L. (2012) ‘Theory of the triple constraint—A conceptual review’, in 2012 IEEE International Conference on Industrial Engineering and Engineering Management. IEEE, pp. 1991–1997.
42. Vung, Pham & Nguyen, Du & Donan, Christopher. (2022). Road Damage Detection and Classification with YOLOv7. 6416-6423. 10.1109/BigData55660.2022.10020856.
43. Wang K, Zhang J, Gao G, et al., 2022. Causes, Risk Analysis, and Countermeasures of Urban Road Collapse in China from 2019 to 2020[J]. Journal of Performance of Constructed Facilities, 36(6):04022054.
44. Witten, I.H., Frank, E., Hall, M.A., Pal, C.J., 2016. Data Mining: Practical Machine Learning Tools and Techniques. Data Mining: Practical Machine Learning Tools and Techniques. <https://doi.org/10.1016/c2009-0-19715-5>.
45. Zhang, A., Wang, K. C., Li, B., Yang, E., Dai, X., Peng, Y., Fei, Y., Liu, Y., Li, J. Q. & Chen, C. (2017), Automated pixel-level pavement crack detection on 3D asphalt surfaces using a deep-learning network, Computer-Aided Civil and Infrastructure Engineering, 32(10), 805–19.
46. Zhang, K., Cheng, H. D. & Zhang, B. (2018), Unified approach to pavement crack and sealed crack detection using pre-classification based on transfer learning, Journal of Computing in Civil Engineering, 32(2), 1–12.
47. Zhang, L., Yang, F., Lv Y, et al. Road Crack Detection Using Deep Convolutional Neural Network with Multiscale Analysis. Sensors, 2022;22(10): 3737.
48. Zhang, Z, Kang J, Li P, et al. Road surface damage detection based on the improved deep convolutional neural network algorithm. Journal of Advanced Transportation, 2021;2021: 1-10.
49. Zhang, L., Yang, F., Zhang, Y. D. & Zhu, Y. J. (2016), Road crack detection using deep convolutional neural network, in Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, 3708–12.
50. Zhang, Z, Kang J, Li P, et al. Road surface damage detection based on the improved deep convolutional neural network algorithm. Journal of Advanced Transportation, 2021;2021: 1-10.
51. Zalama, E., Gomez-Garcıa-Bermejo, J., Medina, R. & Llamas, J. (2014), Road crack detection using visual features extracted by Gabor filters, Computer-Aided Civil and Infrastructure Engineering, 29(5), 342–58.
52. Zhang, A., Wang, K. C., Li, B., Yang, E., Dai, X., Peng, Y., Fei, Y., Liu, Y., Li, J. Q