**Computer Vision for Assembly Line**

**(CVAL)**

**Final Project Plan**

**Team Members:-**

Abhishek J ([aj484@snu.edu.in](mailto:aj484@snu.edu.in))

Akul Dhingra ([ad428@snu.edu.in](mailto:ad428@snu.edu.in))

Anushree Sharma ([as496@snu.edu.in](mailto:as496@snu.edu.in))

Ayush Joshi([aj476@snu.edu.in](mailto:aj476@snu.edu.in))

Harshil K Vora([hv405@snu.edu.in](mailto:hv405@snu.edu.in))

Yash Kumar Singh([ys490@snu.edu.in](mailto:ys490@snu.edu.in))

**TABLE OF CONTENTS**

1. Purpose

2. Revision Chart

3. Software Project Management Plan

3.1 Introduction

3.1.1 Project Overview

3.1.2 Project Deliverables

3.1.3 Evolution of SPMP

3.1.4 Reference Material

3.1.5 Definitions, acronyms, and abbreviations

3.2 Project Organization

3.2.1 Process Model

3.2.2 Organizational Structure

3.2.3 Organizational Boundaries and Interface

3.2.4 Project Responsibilities

3.3 Managerial Process

3.3.1 Management Objectives and Priorities

3.3.2 Assumptions, Dependencies, and Constraints

3.3.3 Risk Management

3.3.4 Monitoring and Control Mechanisms

3.3.5 Cost Estimation

3.4 Technical Process

3.4.1 Methods, Tools and Techniques

3.4.2 Software Documentation

3.4.3 Project Support Functions

3.5 External Interface Requirements

3.5.1 User Interfaces

3.5.2 Hardware Interfaces

3.5.3 Software Interfaces

3.5.4 Communication Interfaces

3.6 Work Elements, Schedule, and Budget

3.6.1 Work elements and Schedule

**1. Purpose**



The Final Project Plan gives information about Team organization, Team leaders, deliverables, Tools, etc. This document is created before the actual project starts and updated throughout the project as tasks are completed and procedures are refined. Final Plan also documents all issues related to client requirements (such as deliverables and acceptance criteria), the project goals, the project organization, the division of labor into tasks, and the allocation of resources and responsibilities.

**2. Revision Chart**



This chart contains a history of this document’s revisions

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Primary Author(s) | Description | Date Completed |
| Draft | Yash Singh, Harshil K Vora | Looking into the types of solutions | 25th August 2015 |
| Preliminary Document | Yash Singh, Harshil K Vora | Finalizing and comparing efficiency of all possible solutions | 27thAugust 2015 |
| Final Report | Akul Dhingra, Abhishek J | Composing core information about SPMP | 12th November 2015 |
| Revision | Yash Singh, Anushree Sharma, Ayush Joshi | Finalizing details and minor corrections | 15th November 2015 |

**3. Software Project Management Plan**



**3.1 Introduction**

This introduction of the Software Project Management Plan (SPMP) shall provide an overview of the project and the product, a list of project deliverables, the plan for development and evolution of the **Computer Vision for Assembly Line (CVAL)**, reference materials for the project, and definitions and acronyms used within the Computer vision for assembly line.

**3.1.1 Project Overview**

The Computer Vision for Assembly Line (CVAL) is a system to provide a computer vision to the die manufactured on an assembly line by client and to detect any defects in the die manufactured. However, CVAL system can be used for any other assembly line product. The system is used to maximize the productivity by providing a system to detect defects in the assembly line manufactured products and ease the workload of the person reviewing the products manufactured.

**3.1.2 Project Deliverables**

|  |
| --- |
| Deliverable |
| Project Plan |
| Feasibility Study Report |
| Software requirement Specification (SRS) |
| Software Design Document (SDD) |
| Software Testing Document |
| Requirement Analysis Document |
| Final Report |
| Traceability Matrix |
| Final Project with Manual |

**3.1.3 Evolution of SPMP**

The SPMP has been updated with completion of each phase by members and updates have been recorded in the document.

**3.1.4 Reference Material**

A concise Introduction to Software Engineering,PankajJalote

Wikipedia([www.wikipedia.com](http://www.wikipedia.com/))

[http://www.utdallas.edu/~chung/RE/syllabus.htm](http://www.utdallas.edu/%7Echung/RE/syllabus.htm)

https://www.utdallas.edu/

**3.1.5 Definitions and Acronyms**

**CVAL: -**Computer Vision for Assembly Line

**Product:** -The manufactured die on assembly line

**SDLC:** -Software Development Life-Cycle

**Agenda: -** It is list of topics to be reviewed, discussed or decided upon at meeting.

**Attendee**: A user who receives an Invite to a meeting, and has the task of responding to the invitation with the pertinent information. An Attendee may or may not attend, and furthermore might be classified as “Important”, “Active”.

**Important participant**: A user who is an Attendee, and is required to attend a meeting.

**Meeting Proposal**: It is an invitation from initiator to potential attendees which includes meeting topic, data range and asks invitees to send their preferences regarding date, place and equipment.

**User**: The person, or persons, who operate or interact directly with the product. The user(s) and the customer(s) are often not the same person.

**Customer**: The person, or persons, who pay for the product and usually (but not necessarily), decide the requirements. In the context of this recommended practice the customer and the supplier may be members of the same organization.

**3.2. Project Organization**

This section of the SPMP shall specify the process model for the project, describe the project organizational structure, identify organizational boundaries and interfaces, and define individual responsibilities for the various project elements

Team Size – 6 members.

Members :-

Abhishek J

Akul Dhingra

Anushree Sharma

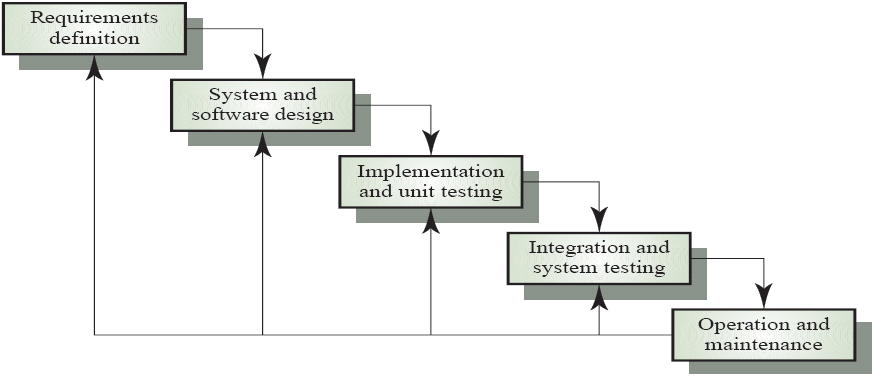
Ayush Joshi (Lead)

Harshil K Vora

Yash K Singh.

**3.2.1 Process Model**

The process model to be used for the earlier phase of the project was the Waterfall Model with the ability to accept change; i.e. we will be able to provide feedback to earlier phases and change it or evolve it for the better based on new information acquired by reviews, comments and also by our own deeper understanding of project.We also kept a possibility for the user to suggest or add in more changes in the software.This model encompasses the following activities: system engineering, requirement analysis, software design, implementation, and testing. The requirement analysis will be highly emphasized. After the software design phase, a mockup will be built to show the functions and operations of the product. Later, an executable product with primary functions will be implemented.

****

Requirement Definitions

System and software Design

Coding and Unit testing

Integration and System Testing

|  |
| --- |
| Done by: Yash Singh, Anushree Sharma, |

|  |
| --- |
| Done By:  Harshil K Vora  Abhishek Jaychandran |

|  |
| --- |
| Done By:  Ayush Joshi |

**3.2.2 Organizational Structure**

For the software development, the task isdivided and allocated among the group members in the beginning

|  |  |
| --- | --- |
| Team Member | Roles |
| Abhishek J | Design Analyst |
| Akul Dhingra | Project planner |
| Anushree Sharma | Requirement Analyst |
| Ayush Joshi | Lead Programmer and Project Head |
| Harshil K. Vora | Design Analyst |
| Yash Kumar Singh | Requirement Analyst |

**3.2.3 Organizational Boundaries and Interface**

The Project lead is responsible for communication between other members for every phase and meetings were held from time to time to have coordination amongst the members and also have an update on every individuals work.

**3.2.4 Project Responsibilities**

All members of the team will be involved in all phases of the project.

## Project Lead

Project Lead is responsible for meeting management, scheduling, agenda, managing project deliverables and their progress, ensuring the submission of project deliverables within the budget and on schedule.

## Requirement Engineer

Requirement Engineer is responsible for: gathering and documenting various system requirements.

**Software Engineer**

Software Engineer is responsible for the development of the system interface mockup and prototype, as well as the User Manual.

|  |  |
| --- | --- |
| **Activity** | **Contributors** |
| Scheduling and keeping track of meetings | Ayush Joshi, Akul Dhingra |
| Preliminary Requirements | Ayush Joshi, Harshil K Vora, Yash Kumar Singh |
| Requirement Analysis and Documentation | Yash Kumar singh, Anushree Sharma |
| Design Analysis and Documentation | Harshil K Vora, Abhsihek J |
| Coding and functional testing | Ayush Joshi |
| Final Testing Documentation | Ayush Joshi, Akul Dhingra |
| Improved understanding of the requirements | Ayush Joshi, Yash Kumar Singh |
| Traceability | Yash Kumar Singh, Anushree Sharma, Abhishek J, Harshil K Vora |
| Updated Traceability | Yash Kumar Singh, Harshil K Vora, Abhishek J |
| Merging Final Document | Anushree Sharma, Akul Dhingra |
| Final Document | All |

**3.3 Managerial Process**

**3.3.1 Management Objectives and Priorities**

Management, which in our case comprises of all members, is responsible for getting activities completed efficiently and effectively with and through each member. The main objective of the management is to organize the meetings for discussions, check the status of the project, and submit the project on time.

## The main objectives are:

* Planning
* Organizing
* Directing
* Coordinating
* Reporting

**3.3.2. Assumptions, Dependencies and Constrains**

Assumptions made for the project are:

* The difficulties faced with any task assigned shall be reported immediately.
* Customer will have the memory buffer on which image storing and retrieval operations as well as product quality log storage will be conducted.
* Customer will not make any frequent changes in requirements.
* Each team member having expertise in certain field will help others to achieve better and faster results.

Project Constrains are:

* **Time:** The time frame during which all the activities should be completed and a final software is developed.
* **Quality:** The quality that the project should have so that it doesn’t fail/become obsolete in near future.
* Possibility of members not able to attend scheduled meetings.
* Possibility of lack of resources to complete the software in time.

**3.3.3 Risk Management**

Risk management is the identification, assessment, and prioritization of risks followed by coordinated and economical application of resources to minimize, monitor, and control the probability and/or impact of unfortunate events. Any changes made in the organization which contradicts the assumptions made can be accommodated with minimal changes in the code.

Possible risks and mitigation or avoidance strategies:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risk** | **Possibility** | **Impact** | **Description** | **Solution** |
| 1.Disk Failure | Low | High | Machine having all project deliverables has failure. | The project deliverables and documents will be stored in each member’s laptop. |
| Lack of skill | High | High | Team member is not familiar with tools or techniques used in this project | Platform and language is chosen such that most of the members are comfortable with it. |
| Poor Quality | Medium | Medium | Team is not able to reuse code and/or documentation. The software becomes obsolete due to negligence while software development | Holding meeting timely will ensure that each member is doing his part in the given time frame and project is performing efficiently and properly. |
| Completion in time | Medium | High | A running software is not developed in given time frame. | We have developed a plan and a schedule which will be followed strictly. |
| Lack of communication | Low | High | Team members fail to communicate at critical times | Monitor team member response and project involvement. Provide friendly environment. |

**3.3.4. Monitoring and Controlling Mechanism**

* Each phase will have at least 2 members.
* The members are responsible for completion of the phase.
* The lead will provide details of each member’s responsibilities, due dates for completion, meeting time and agenda and review of final document.
* Meetings will be set up to review key components at every stage.
* Documentation will also be updated after each meeting.

**3.3.5 Estimation plan**

A cost estimate is the approximation of thecost of a program, project, or operation. The cost estimate is the product of the cost estimating process. The cost estimate has a single total value and may have identifiable component values. The cost estimation plan for our project is done using function pointer and the estimation of our project has been separately documented.

**3.4 Technical Process**

**3.4.1 Methods, Tools and Techniques**

The following tools are used for development of software:

* C++ (OpenCV) is used for coding.
* Documentation will be done using Microsoft Word.
* Minutes of meeting will be deposited in an Excel sheet.
* Whatsapp Group/Gmail will be used to communicate between Members

**3.4.2 Software Documentation**

* The following software documents will be developed:
* Software Project Management Plan (SPMP).
* Software Requirement Specification (SRS)
* Software Design Document (SDD)
* Software Testing Document
* Function Point Estimation
* Requirement Analysis Document
* Traceability Matrix
* Minutes of Meeting

**3.4.3 Project Support Function**

* Quality Assurance
* Verification and Validation
* Training
* Configuration Management

**3.5 External Interface Requirements**

**3.5.1 User Interfaces**

* User will use a camera to capture an image of the product that comes from assembly line. Images are clicked in different angles.
* In case of High defect density, the assembly line engineer will be contacted immediately.

**3.5.2 Hardware Interfaces**

* Camera for picture input.
* A designated Computer to Run the Software in.
  + 1. **Software Interfaces**
* **C++(Open CV):** Open CV module is used in C++ to implement image processing operations.
* **Visual Studio Community:** Used as an IDE for development of code
  + 1. **Communication Interfaces**
* Computer Vision for Assembly Line (CVAL) will not directly interface with any communication system.

**3.6 Work Elements, Schedule and Budget**

* + 1. **Work elements and Schedule:**

This project is scheduled to be completed by November20, 2015 for the final demo. Here is the outline of the timeline of the deliverables:

* Preliminary Project Plan to be prepared by 25th August 2015.
* Feasibility Study Report should be prepared by30th August 2015.
* Final Project Plan shall be prepared by 10th September.
* Software Requirement Specification (SRS) should be completed by 20thSeptember 2015
* Software Design Document(SDD) to be completed by 5th October 2015
* Software Testing Document shall be prepared by 1st November2015
* Final Traceability Matrix to be completed by 10th October 2015
* Final Report to be completed by 15th November 2015.

**Project Definition**



The Computer Vision for Assembly Line(CVAL) is a system to provide a computer vision to the die manufactured on an assembly line by client and to detect any defects in the die manufactured. However, CVAL system can be used for any other assembly line product. The system is used to maximize the productivity by providing a system to detect defects in the assembly line manufactured products and ease the workload of the person reviewing the products manufactured.

**Existing System**



Currently there is no such type of software available in the industry. All the work is done manually. The current system works in following manner:

* There is no provision for finding defects in the produced goods through a software. Manual Labour is required for the purpose.
* No log is maintained about the products which are passed and found defective.

**Disadvantages**



Here are some of the disadvantages of the current system:

* Defects are found manually which are not very reliable.
* Since the defects are found manually, it takes a lot of time to detect defects for each produced good.
* No logs are maintained and thus makes it hard to keep a track of number of defective produce and judge if the machine requires some repairs.

**Solutions Discussed**



1. Using lasers/ultrasound to detect if the material is defected

2. Using a live video feed from a webcam to detect if the material is defected.

3. Machine learning Approach/contour tracer.

**Proposed Solution**



System shall provide facilities to reduce the problems in current system. Here are some facilities that system will provide:

* A software to detect the defects in the produced goods, if any.
* A log maintained for the produces.
* The assembly line engineer is contacted immediately in case of high defect density.
* The software detection shall maximize the productivity by providing a system to detect defects in the assembly line manufactured products.
* Software will ease the workload of the person reviewing the products manufactured.

All these facilities will help in reducing time and effort of the people working in the field.

**Feasibility Analysis**



**Why the Study was undertaken?**

The feasibility Study has been undertaken due to sole purpose of determining and devising a solution for the problem stated above within the given frame, manpower and resources available.

**Feasibility Aspects:**

**1. Financial Analysis:**

* Shall be used in assembly lines to detect defective objects (cracks, improper dimensions, etc.)
* Returns shall be guaranteed as industries today rely heavily on assembly lines.
* Cheapest Alternative thus high return on investment.

**2. Technical Feasibility:**

For development of any application its technical feasibility has to be done. We are planning to develop a software that can be implemented for any Assembly Line Product and hence there are many technical constraints which should be kept in mind. We shall compare the available technologies in the market for this system. Since its system based software; the system will contain C++(Open CV) IDE, a camera input and some memory buffer software from customer end to manage the log. The choices for the code development tools were as follows:

Java: As the system is required to process images in an efficient manner, Java is not efficient as it doesn’t have any inbuilt functions for the software and Java is more apt for Object Oriented Programming which is not required in our Software.

Python: Python was more apt but given the nature of the project which requires fast image processing lack of unique functions for the same and a very minimal knowledge of the language amongst the group members made us eliminate this choice.

C++(OpenCV): We chose OpenCV as there are inbuilt functions for image processing which makes it best suitable for the software. OpenCV Libraries are most compatible with C++ as well as the fact that C++ is a lighter language than Java or Python which aim for cross platform operability and ease of code respectively, whereas the main environment for this program Is a factory main-frame hence smaller code, working on singular environment is preferable.

Hence we concluded to code the software using C++(OpenCV).

**3. System Requirements:**

Fast processing speed, Medium-Large memory, connections (High speed) to memory buffer and required Hardware.

**Project Impacts**



The project is being developed keeping in mind the current problems being faced in the assembly line, detection of defects, maintenance of log records for each produce which requires a lot of human effort to get the job done. Since minute defects also cannot be left out, manual method of detection is not very reliable. Once the software implementation for finding defects is done, the entire Process(s) shall become easier, efficient and hassle-free. Apart from defect detection, a log shall also be maintained, keeping a record of all the defective produces which will help customer keep a track.

**Conclusion**



After discussing all the requirements, solutions and possible alternatives, it can finally be concluded that such a system is feasible and also required in order to improve the existing process and system of defect detection in the assembly line and endorsing this project would be beneficial for all the people working in assembly line.