

King Fahd University of Petroleum and Minerals
Department of Systems Engineering

SE 490 - Senior Project Design

Design of a Rotational Speed Measurement System by Computer Vision for Quality Testing

Project Team

Ibrahim Al-Safadi 227496

Mohammad Shahab 227598

Project Advisor

Dr. Sami El-Ferik

Abstract

An in depth description of an engineering design process for an industrial problem related to quality control is offered. The design is related to developing an automatic rotational speed measurement system utilizing Computer Vision and Image Processing techniques. A comprehensive analysis and evaluation of the designed system is also conducted.

19 August 2007

ACKNOWLEDGEMENT

We would like to thank the senior project advisor, Dr. Sami El-Ferik, for his continuous support and consulting even through his vacation.

The System Engineering Department is also thanked for providing equipment and technical support via its lab engineer, Mr. Shaukat Ali.

We would also like to thank Dr. Chokri Belhaj Ahmed from the Electrical Engineering Department for his assistance.

Mr. Amrou Al-Sharif is also thanked for his suggestions and remarks.

Finally, we would like to express our gratitude and appreciation to all of whom provided support to us throughout our undergraduate years.

TABLE OF CONTENTS

TABLE OF CONTENTS.....	2
LIST OF FIGURES	5
LIST OF TABLES	7
REPORT ORGANIZATION	8
1. PROBLEM STATEMENT	9
1.1 SUMMARY	9
1.2 PHYSICAL SYSTEM DESCRIPTION	9
2. PROJECT PLANNING	11
2.1 LITERATURE SURVEY ABOUT THE PROBLEM	11
2.2 SEARCHING FOR ALTERNATIVES	12
2.2.1 Contact vs. Non-Contact Rotational Speed Sensors.....	12
2.2.2 Instantaneous vs. Average speed Measurement.....	14
2.2.3 Physical form of the sensor.....	16
2.2.3.1 Electromagnetic (Non-Contact).....	16
2.2.3.2 Optical Tachometers.....	17
2.3 JUSTIFICATION OF VISION.....	18
2.4 PROJECT OBJECTIVES.....	22
3. DESIGN PROCESS.....	23
3.1 EQUIPMENT SEARCH / DATA COLLECTION	23
3.1.1 Image Acquisition: Camera.....	24
3.1.1.1 Digital Imaging Technology Concepts.....	24
3.1.1.1a Image Sensors.....	25
3.1.1.1b Scanning Formats.....	26
3.1.1.1c Frame Rate.....	27
3.1.1.1d Communication.....	28
3.1.1.2 Cameras in Market.....	30
3.1.1.3 Frame Rate Analysis	34

3.1.1.3a	Frame Rate for Mark Extraction	35
3.1.1.3b	Frame Rate for Edge Detection	36
3.1.1.4	Project's Camera.....	38
3.1.2	Fan	42
3.1.2.1	Search Process	42
3.1.2.2	Selection and Solution.....	45
3.1.3	Photo Tachometer	47
3.1.4	Desktop Computer	49
3.1.5	Other Equipment	50
3.1.5.1	Continuous Voltage Transformer	50
3.1.5.2	Metal Mounting	51
3.1.5.3	Trip Switches	51
3.2	IMAGE PROCESSING BACKGROUND	53
3.2.1	Digital Image Processing: Introduction.....	53
3.2.2	Feature Extraction.....	59
3.2.2.1	Methods	59
3.2.2.2	Edge Detection.....	61
3.2.3	Morphological Image Processing.....	66
3.2.4	Computer Vision and Machine Vision	69
3.3	ALGORITHM AND SOFTWARE DEVELOPMENT.....	72
3.3.1	Software Used.....	72
3.3.1.1	LabVIEW	72
3.3.1.2	NI Vision Assistant.....	74
3.3.2	Attempted Algorithms.....	75
3.3.2.1	Mark Extraction	75
3.3.2.2	Edge Detection.....	79
3.3.3	The Implemented Algorithm	79
3.3.3.1	Step 1: Frame Capturing	79
3.3.3.2	Step 2: Time Stamping	81
3.3.3.3	Step 3: Frame Retrieving & Zooming	81
3.3.3.4	Step 4: Edge Detection	83
3.3.3.5	Step 5: Extracting Edge Information	85
3.3.3.6	Step 6: Speed Calculation.....	87
3.3.3.7	Step 7: Quality Decision	88
3.3.4	Improved Algorithm.....	89
3.3.4.1	Algorithmic Options.....	89
3.3.4.1a	Method 1: Angle Measurement	89
3.3.4.1b	Method 2: Latching Coordinates.....	91
3.3.4.2	Implemented Improved Algorithm	92
3.3.4.2a	Step 5: Extracting Edge Information	92

3.3.4.2b	Step 6: Speed Calculation.....	94
3.3.5	Software Outputs	96
3.3.5.1	User Interface	96
3.3.5.2	Automatic Decision	100
3.4	PHYSICAL SYSTEM DESIGN.....	101
3.4.1	Time Synchronization.....	101
3.4.2	Camera Mounting.....	102
4.	RESULTS AND ANALYSIS	103
4.1	ALGORITHM TUNING	103
4.1.1	Parameters	103
4.1.2	Zooming Area.....	104
4.2	EXPERIMENTAL AND STATISTICAL ANALYSIS	105
4.2.1	Collecting Samples	106
4.2.2	Distributions and Statistical Analysis	108
4.2.3	Output Statistical Analysis	112
4.3	MEETING OBJECTIVES	113
5.	CONCLUSION.....	115
5.1	PROJECT INNOVATION.....	115
5.2	APPLIED RECENT TECHNOLOGY	116
5.3	RECOMMENDATIONS	117
5.4	REPORT CONCLUSION	119
	REFERENCES.....	120
	APPENDIX A: LABVIEW FULL PROGRAM	121
	APPENDIX B: FULL STATISTICAL ANALYSIS OUTPUT	126
	APPENDIX C: FULL LIST OF DATA COLLECTED.....	136
	APPENDIX D: ATTACHED CD CONTENTS.....	143

LIST OF FIGURES

FIGURE 2.1 ELECTRICAL TACHOMETER.....	13
FIGURE 2.2 ORIENTATIONS BETWEEN FAN AND CAMERA SURFACES.....	20
FIGURE 3.1 A CCD SENSOR CHIP.....	25
FIGURE 3.2 POINT SCANNING & LINE SCANNING	27
FIGURE 3.3 FIREWIRE PLUG	29
FIGURE 3.4 USB PLUG	30
FIGURE 3.5 CAMERA EXAMPLE FROM ALLIED VISION TECHNOLOGIES	33
FIGURE 3.6 CAMERA EXAMPLE FROM BASLER	34
FIGURE 3.7 FRAME RATE ANALYSIS.....	35
FIGURE 3.8 SLIM 320 IN LAB.	40
FIGURE 3.9 FAN SELECTED.....	46
FIGURE 3.10 HAMPDEN HPT-100A IN OPERATION	48
FIGURE 3.11 CONTINUOUS TRANSFORMER FROM BACK.....	51
FIGURE 3.12 STEPS OF DIGITIZING AN IMAGE.....	54
FIGURE 3.13 IMAGE MATRIX	56
FIGURE 3.14 EDGE MATHEMATICAL ANALYSIS	63
FIGURE 3.15 EXAMPLE OF FULL IMAGE EDGE DETECTION	65
FIGURE 3.16 IMAGE & VISION RELATED RESEARCH TOPICS TREE.....	69
FIGURE 3.17 VISION ASSISTANT SCREENSHOT.....	74
FIGURE 3.18 MARK DETECTION. ABOVE: ORIGINAL IMAGE. BELOW: EXTRACTED MARK.....	78
FIGURE 3.19 TIME STAMPING LABVIEW BLOCK DIAGRAM	81
FIGURE 3.20 RETRIEVING SAVED IMAGE FRAMES LABVIEW BLOCK DIAGRAM.....	82
FIGURE 3.21 ZOOMING AREA LABVIEW BLOCK DIAGRAM.....	83
FIGURE 3.22 EXAMPLE OF EDGE DETECTED	84
FIGURE 3.23 EDGE DETECTION LABVIEW BLOCK DIAGRAM	85
FIGURE 3.24 EXTRACTING EDGE INFORMATION LABVIEW BLOCK DIAGRAM.....	86
FIGURE 3.25 SPEED CALCULATION LABVIEW BLOCK DIAGRAM	88
FIGURE 3.26 QUALITY DECISION LABVIEW BLOCK DIAGRAM.....	88
FIGURE 3.27 ANGLE MEASUREMENT ANALYSIS.....	90
FIGURE 3.28 IMPROVED SPEED CALCULATION LABVIEW BLOCK DIAGRAM.....	95
FIGURE 3.29 'OPERATION' PAGE OF USER INTERFACE.....	97

FIGURE 3.30 'ADVANCED' PAGE OF USER INTERFACE.....	98
FIGURE 4.1 ILLUSTRATION ABOUT REJECTING OUTLIERS	112

LIST OF TABLES

TABLE 3-1 LIST OF DIFFERENT CAMERAS	31
TABLE 4-1 FEW SAMPLE DATA.....	107
TABLE 4-2 MEAN AND VARIANCE VALUES.....	111

REPORT ORGANIZATION

This report is divided into five major sections of variable length. It starts with an in-depth description of the industrial problem, followed by a section on the project plan including literature survey and searching for alternatives. The method selected by the team is elaborately justified in this section as well.

The third section is the lengthiest and, certainly, the most important, presenting the search for equipment, a prelude on the image processing techniques used, and the software development along with its graphical user interface. The fourth presents analysis and experimental results of the output.

The report is concluded in the last section where recommendations and suggestions are made for further extensions and continuing projects.

1. PROBLEM STATEMENT

1.1 Summary

A local industrial plant specialized in manufacturing air conditioners requests a solution to a quality control problem. At a certain phase near the end of the manufacturing process, the AC fans are assumed to rotate at a specific speed range. This speed is to be measured while maintaining a few conditions and offering the highest level of autonomy possible with no interference (contact) with the process.

Main difficulties confronting this task include the continuous motion of the conveyer belt, time limitation, cost, reliability, and labor requirement.

1.2 Physical System Description

The air conditioners are transported through a testing lab by a conveyer belt, which is in continuous motion.

The fans have four blades colored black and are made of plastic. The part of the air conditioners where the fans are mounted is at the opposite part from where the technician stands.

At this phase of the manufacturing process there is a partial view of the fans from the left and upper sides of the air conditioners.

- Quality Test Range: 800 – 2000 RPM
- Material of fan: metal
- Blades Color: Black
- Conveyor Belt Average Speed: 0.5 ft/s

2. PROJECT PLANNING

2.1 Literature Survey about the Problem

In attempt to find a good solution to this problem, a broad search in literature was conducted via the internet. It was noticed that technical papers and previous efforts to measure rotational speed are rare in the literature. While a vast amount of papers and experimentation reports were found related to linear speed measurement by image processing, most particularly vehicle speed (refer to “Video Image Processing” by Dailey and Li from Uni. Of Washington), a very few amount of papers were concerned with angular speeds per se. Nevertheless, useful information was obtained from the former papers, especially in the common problem of moving edge detection.

The most significant papers found were the result of research conducted in Ishikawa Namiki Komuro Laboratory in the University of Tokyo, Japan. However, the emphasis in their research was not on the image processing part specifically, but rather on what they called “multi-target tracking” for high speed vision chips. The part on rotation measurement was merely an application to their main research results.

Even so, their methodology for detecting speed relied on advanced analysis of vector distributions from the marks (cavities) on the rotating golf ball, a technique certainly not in our possession:

“[The] measurement estimates the rotation information from the distribution of speed vectors on a sphere. The vectors are obtained by multi-target tracking. The method includes three processing steps as follows: (1) Obtaining two-dimensional trajectories of some regions on a sphere. (2) Determining three-dimensional speed vectors. (3) Estimating rotation information.” [Yoshihiro Watanabe]

2.2 Searching for Alternatives

For generality and unbiased preliminary conclusions, the search was not initially confined to specific solutions, but rather, all methodologies for rotational speed measurement were studied regardless of whether they seemingly violated the constraints posed by the problem at hand or not. It was found that rotational speed measurement systems can be categorized in several manners, of which these are probably the most important:

2.2.1 Contact vs. Non-Contact Rotational Speed Sensors

In this categorization scheme, the matter of whether the sensor is in physical contact with the process or not is critical. Contact rotational speed sensors

are either electrical or mechanical based. According to Britannica Encyclopedia:

“Mechanical tachometers utilize the fact that the centrifugal force on a rotating mass depends on the speed of rotation and can be used to stretch or compress a mechanical spring. A resonance, or vibrating-reed, tachometer uses a series of consecutively tuned reeds to determine engine [or motor] speed by indicating the vibration frequency of the machine.

Electrical tachometers are of several types. The eddy-current, or drag, type is widely used in automobile speedometers; a magnet rotated with the shaft being measured produces eddy currents that are proportional to angular speed. Electric-generator tachometers work by generating either an alternating or a direct current.” [Encyclopaedia Britannica]

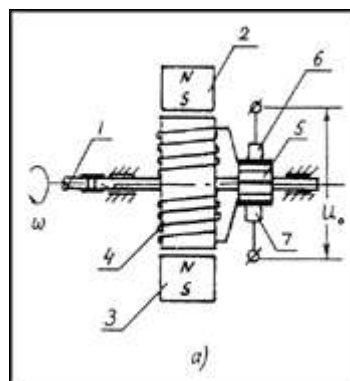


Figure 2.1 Electrical Tachometer

However, electrical based sensors are not limited to contact measurements. In fact, most non-contact rotational speed sensors rely on electricity in one way or the other. (By electrical we mean those which depend on physical variables such as proportional voltage and current, as opposed to

computational procedures like vision and image processing). Non-contact rotational speed sensors include laser-based, optical reflective, electromagnetic flux, and vision-based measurement systems.

It was noticed from product survey on the internet that contact sensors have significantly smaller speed measurement range when compared to non-contact sensors. Several websites offered dual devices that contained both types, with the non-contact sensor having a capability to measure rotational speeds up to 100,000 rpm, while the contact sensor measuring only up to 20,000 rpm.

Regarding the industrial problem at hand, there was no doubt that the non-contact type of sensing scheme was the way to go. Even though the contact sensors were more than enough in terms of speed measurement range, the position of the fans in the air conditioners at the testing phase prohibits access to the rotating axle. Furthermore, automating the measuring processes in this case would be very difficult.

2.2.2 Instantaneous vs. Average speed Measurement

Although not of much practical concern here, this categorization is theoretically significant and is vital for some applications too. Instantaneous sensors are those that depend on proportionality to continuous physical variables, and hence can give instantaneous readings.

Britannica Encyclopaedia restricts the term ‘tachometer’ to this type of instantaneous measurement only. Hence, for example, an electromagnetic sensor that produces eddy currents proportional to magnetic flux variation can be labeled under the ‘tachometer’ group, while an electromagnetic based sensor that depends on processing the flux variation into a counting scheme is not considered a tachometer.

The other type of measurement system relies on counting the repetition of a certain event (e.g. crossing of an object, reflection of a laser beam, etc.) and comparing it to a time reference. The output is not an instantaneous speed but rather an average speed in reference to a time interval. Although this may seem to be less accurate than the former type, it still can approach the instantaneous measurement by shrinking the differential interval if the computation is fast enough. It also might be more appropriate for processes that don’t require a high level of accuracy.

Vision based speed measurement is essentially differential. Hence it can be claimed, since the time interval between two frames is not infinitesimal, that its output measurement is an average by nature. This is definitely the case in our project as we relied on measuring time difference between specific frames to determine speed.

2.2.3 Physical form of the sensor

This categorization scheme considers the physical nature of the elementary sensing object in the measurement system. To avoid repetition and redundancy, only sensing devices of direct consideration to this project will be mentioned:

2.2.3.1 Electromagnetic (Non-Contact)

In this type of sensing, the electromagnets are positioned orthogonal to the rotating object with no mechanical parts or coils in contact. As the sensor emits electromagnetic waves near the rotating object, a variable flux generates perpendicular to the rotational axis of the fan as it revolves.

This variability can be processed and then, by an algorithm, converted to a counting device which will eventually become a speed sensor when counting is associated with a timer.

The main advantages of this type include absence of external load on the rotating object, high precision, and extended life time of the sensor as it is free from wear caused by friction.

On the practical side, however, even though automating the measurement is simple in this case after obtaining the output voltage or current, it seemed that this type of sensing is difficult to implement and inefficient to use in this context. It was found that these sensors are used more frequently in flow rate measurement by vortex detection, where the electromagnets can be uninterruptedly positioned on the external sides of the duct. Obviously, this positioning is not admissible in the situation at hand.

One fact remains to be mentioned, the team has studied previously that some electromagnetic sensors don't work except with metal objects. This rules out the fans in this project which are made of plastic.

2.2.3.2 Optical Tachometers

Sensors that rely on infrared or laser beams share the common physical properties of transmission and reception of rays and depend on photoreceptors to trigger a counter. A few general categories must be distinguished, however.

Reflective vs. Tx./Rx.: The first significant distinguishing factor is whether the optical sensor is composed of one or two separate parts. The second type requires a transmitter that continuously sends a constant beam while the receiver detects an impulse pattern depending on the blockage of the rotating object in between.

The first type is easier to position since the rotational object doesn't have to be surrounded from the two sides by sensing elements, which could be prohibited anyway by the nature of the rotating object. However, in some cases, the reflective sensor requires attaching a reflective sticker on the rotating object, which renders the sensor inconvenient for processes that don't have such stickers.

2.3 Justification of Vision

The team was informed by several professionals that optical sensing based instruments were the mainstream measurement systems for rotational speed measurement in industry. They were also confronted by serious inquiries and demands to justify the reliance on vision and image processing, a no doubt complicated and demanding method, to tackle an industrial problem that has been already solved by easier alternates.

Apart from the essential purpose of relying on vision (i.e. avoiding any contact with process, including marks), the following additional arguments can be made:

The answer to the inquiries is mainly twofold. The first reason is purely experimental and was motivated by seeking innovative procedures. Since one of the main objectives of senior projects is to stimulate students' abilities to design new systems, and tackle industrial problems from new angles and previously untaken perspectives, the team was keen on selecting a

methodology that fulfilled this objective. In short, lack of direct applicability to industry or existence of substitutes and pre-implemented solutions are not legitimate constraints to innovative senior projects.

The second reason has to do with the method itself, namely, vision and image processing. Image processing techniques are very different in form than all other sensing schemes encountered while working on this project. Admittedly, its characteristics sometimes cause it to fail where other sensing schemes excel, while in other cases causing it to outperform all other schemes. What follows is a point-by-point argument clarifying what it has to offer and what it has to be cautious about:

The method requires a high performance camera and PC. However, as an *automatic* measurement system was required from the start, it appears that all other sensing schemes will eventually require a computer or other processing device in case autonomy was in demand. For example, if an optical tachometer was to be integrated into an automatic testing system, it would most likely require functions like timing or sensing for arriving ACs, recording data, making decisions of passage and failure, and probably even—in a higher level context—be linked to the plant's field bus or industrial network which can communicate statistics to the management and manufacturing divisions. The price paid for autonomy no doubt includes a computing system, and if it were not for seeking automation, introducing a camera into the field would be financially unworthy.

The method is highly sensitive to noise and industrial parameter variations. This is certainly true, but is *heavily dependent on the sophistication of the image processing algorithm*. For example, a basic image processing algorithm gives different results when the lighting on the surface of the blades is changed in unequal proportions. This generally occurs when the angle between the fan and camera planes is changed, aside from the trivial source of error resulting when obstacles get in the way. The figure below shows the three degrees of freedom of which the two planes (pertaining to the fan and camera) can move in three dimensional space. Three angles, similar to pitch, roll, and yaw of aircraft are present.

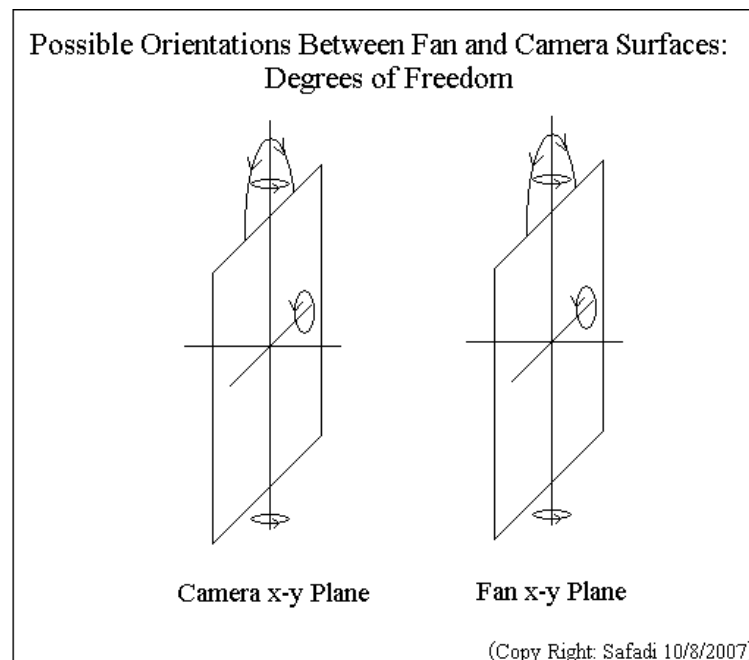


Figure 2.2 Orientations between Fan and Camera Surfaces

Nonetheless, the intelligence of the algorithm can be increased. Advanced image processing techniques can be designed to distinguish shadow, glare and variable background from actual blade edges. They can also be designed

to adapt to angle variations between camera and fan planes. Furthermore, pattern recognition techniques can also be included to enable the computer to recognize the center of the fan (or any other of its components) to help standardize the measurement process.

A final point must be noted here. Research in the applications of image processing revealed the well suitability of its results in *inspection*. In fact, National Instruments has developed software built on the LabVIEW platform for this specific application in industry. Illustrations such as counting parts of certain geometric size or shape, recognizing a defected part by color, determining the inclination degree of objects, and separating parts from defined characteristics identifiable by vision were all observed. Hence it would be wise to exploit this applicability in a fully integrated testing system. This suggestion is further extended and elaborated in the Recommendations part of the report.

2.4 Project Objectives

In light of the senior project proposal and the data collected, the team distinguished four significant objectives to be accomplished. These are as follows:

- 1) Design of a non-contact automatic rotational speed measurement system of a fan.**
- 2) Engineering Analysis and Evaluation of the Measurement System.**
- 3) Development of a software package containing the previous elements.**
- 4) Explore for new applications of image processing.**

3. DESIGN PROCESS

3.1 Equipment Search / Data Collection

In this section of the report, Equipment used during the work of the project will be discussed. Hardware related to the project will be reviewed. There are two types of hardware that is used: 1) hardware which will form the end-product of this project and used during the design process; and 2) hardware which is used only for prototyping and experimentation (i.e. will not be included in the end-product).

The End-Product **hardware** contains: an image acquisition device (i.e. a camera), Desktop Computer and the required integration and communication means. For only the purpose of experimentation and prototyping in lab environment, equipment include: a fan, reference measurement device (i.e. optical tachometer) and other hardware required during this phase.

A detailed information and work of hardware is explained in the following sections.

3.1.1 Image Acquisition: Camera

The camera is the most important device related to the project. The camera stands as the image acquisition device of the fan motion. In this section, a review of the concept of video/image capturing will be discussed. Different video camera products and brands will also be reviewed and characteristics of cameras with relation to the project goals. The section will be concluded by the selection decision.

3.1.1.1 Digital Imaging Technology Concepts

This section will talk about the technology of digital imaging nowadays. Specifically, the section will discuss technology related to professional digital imaging with its applications in many fields. The following paragraphs will focus on technical issues regarding cameras (i.e. video camera) and physical work of them.

When talking about video cameras, principle of work differs, in some way, than of still photography imaging. Video acquisition devices (cameras) capture continuous incoming photo frames of their field of view in the environment in front of them. This mechanism depends on many features of every camera device. Some concepts of these features will be explained in sections following this paragraph, namely, image sensors, image scanning format (and resolution), frame rate and image data communication. Other concepts are also found in different cameras, but are less significant to discuss in the scope of this project application.

3.1.1.1a Image Sensors

Image sensors are the electronic parts that are first in line in front the required view to be captured. They are facing the coming light from environment. They receive the flow of photons and convert it to electric signal. There are two techniques of image sensing. The two technologies used are: 1) Charged-Coupled Devices or simply **CCDs** and 2) Active Pixel Sensors or sometimes called **CMOS Sensors**.

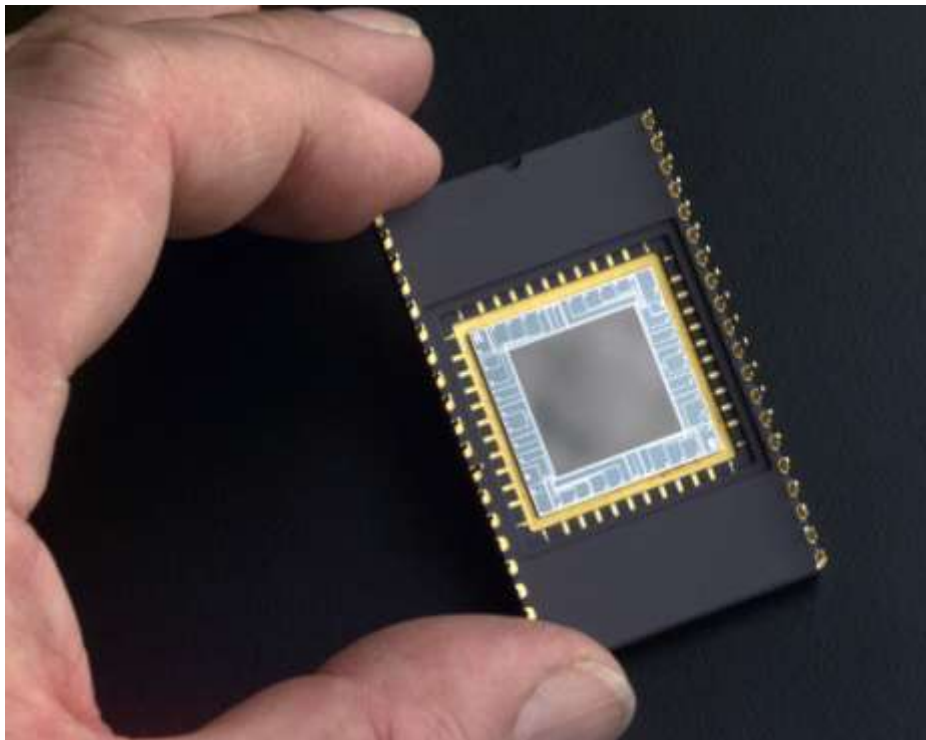


Figure 3.1 a CCD Sensor Chip

Both types work in the same principle of converting light to and electric charge, but differ in other concepts as follows. *Both* sensors work as array (pixelated) of metal oxide semiconductors (CMOS!) receiving incoming light, specifically, photodetectors. Each semiconductor device in the array

builds up an electric charge in proportion of intensity of illumination hitting each one. In CCDs, all charges are collected sequentially to neighboring circuitry to convert the electric charges into voltage signal. However in CMOS sensors, conversion of charges into voltage is initially done locally in every pixel in the array and then transfers the overall output. Analog signal of voltage is then digitized, i.e. Analog-to-Digital conversion, in separate circuit. Each architecture has its own implications. Although both technologies produce the same image quality, CMOS sensors are considered better in the sense of system integration (i.e. System-on-Chip). [Litwiller]

3.1.1.1b Scanning Formats

Scanning format means how the electronic circuit, which is explained in the previous section, is collecting the signal (image). The collecting process reflects the organization of the array of the semiconductor detectors inside the image sensor used. There are 3 kinds of scanning, namely, 1) Point Scanning, 2) Line Scanning and 3) Area Scanning.

In *point scanning*, pixels are sequentially scanned one by one at a time to detect the whole two-dimensional view captured. This technique brings higher resolutions, but system complexity rises as the movement and repetition of scanning occurs. The other technique is to scan in only one-direction one-dimensional line of sensors, which is called *line scanning*. At a time, one line of the view is captured by the sensors before continuing to the next line. The line length depends only on the physical design of image sensors used. Line scanning is faster and simpler than point scanning, but

the resolution is limited by the size of the line. *Area scanning* lets the required view to be captured at one time (single exposure). This is done by a collection of sensors put in a 2D array structure. Area scanning produces the highest speed, but the resolution is limited by the two directions. [Eastman Kodak Company]

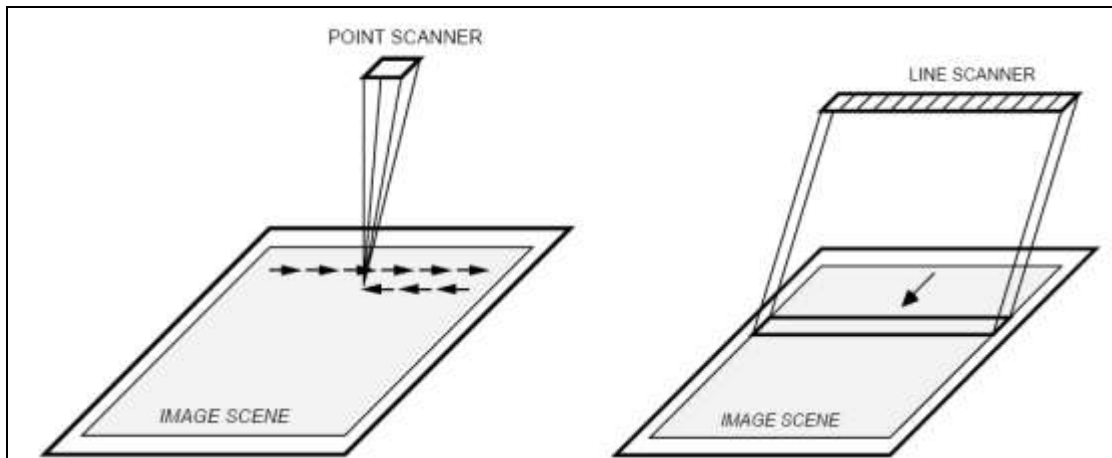


Figure 3.2 Point Scanning & Line Scanning

3.1.1.1c Frame Rate

Another feature of a camera is the time required to acquire the image and completely transfer image data to next stage out of the imaging device. Frame rate is defined as the reciprocal of this time, computed usually in Frame per Second (FPS) or Hertz (Hz). This time involve *two* phases. One phase is the time required to read out the signal from the charge conversion and digitize it for a single pixel, called read time, t_R . Another phase is the exposure time of the capturing itself plus the minor mechanical delays associated, t_{ex} . So, the frame rate of any camera can be approximated by:

$$FPS = \frac{1}{(t_{ex} + N \cdot t_R)}$$

As N corresponds to number of pixels to be read. [Fellers and Davidson]

So for a designated frame rate provided by manufacturers, the number will not necessarily reflect good image quality because the number is a combination of two elements, exposure time and read time. Exposure times of each camera have wide range and depend on the application (e.g. industrial, scientific or artistic photography). Exposure time could range and be controlled from 1 second to 0.1 millisecond for some cameras.

3.1.1.1d Communication

After “acquiring” the image, it is ready to be transferred for next stages, processing or screen view for example. Most of today’s camera systems communicate digitally, especially for processes related to a computer usage.

Image data communication technologies take place in different forms depending on application and other reasons. Transferring image data can be accomplished using technologies like USB, FireWire, Serial, DVI or still Analog.

FireWire cameras use the IEEE.1394 standard of audio and video transmission. FireWire video cameras can be found in industrial, medical, scientific, astronomy and microscopy applications. For this kind of

applications, FireWire camera's interface differs from those of other lower-significance applications. It uses a protocol called DCAM or sometimes called **IIDC** (Instrumentation and Industrial Digital Control) which is set by 1394 Trade Association. This protocol defines that the video camera has an output of uncompressed image data with no audio. A computer that accepts FireWire 1394 protocol will offer a transfer speed up to **800 MBits per second**. 1394 protocol is recommended for industrial control and instrumentation. For more information about the IEEE.1394 standard, visit www.1394ta.org. [Purcell]

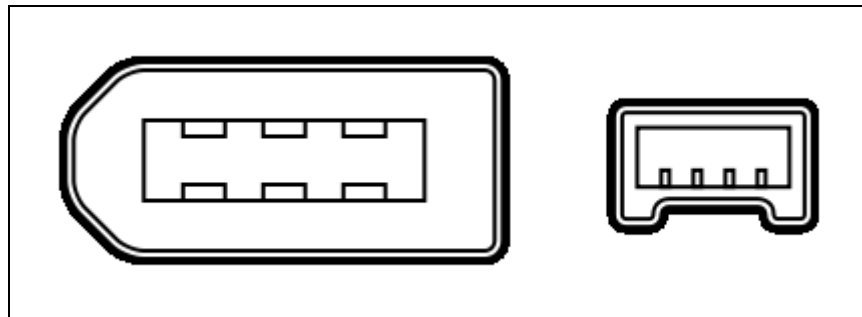


Figure 3.3 FireWire plug

USB, which stands for “Universal Serial Bus”, is used also in camera technology. With today's infamous USB applications associated in everyday office and home life, interface technology of it provides speeds up to 480 MBits per second (for a USB 2.0), but, of course, video applications are limited to internet applications and digital non-professional cameras. [Universal Serial Bus - Wikipedia, the free encyclopedia]

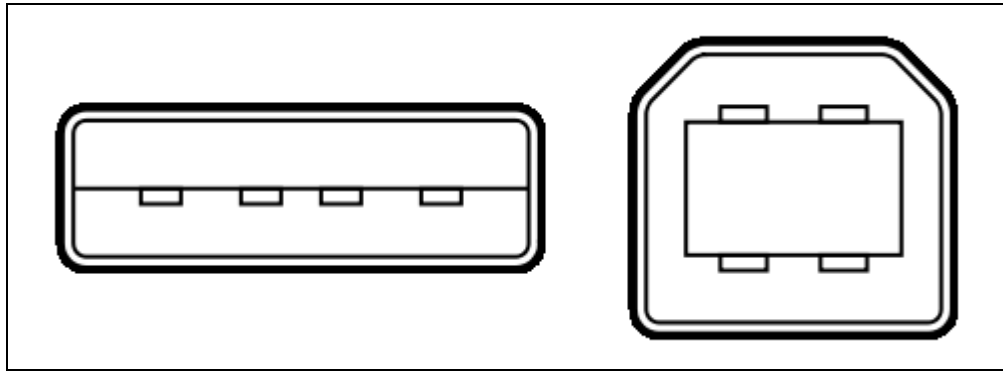


Figure 3.4 USB plug

CamerLink, another common communication method used especially in computer vision applications.

Other means of communication (Analog, DVI or other Serial methods) often occur in specific applications of Display, namely, TV, Projectors and Computer Displays (LCD) which all are not related for computer integration with the camera.

3.1.1.2 Cameras in Market

In this section, different kinds of camera systems will be discussed. The review is going to talk about camera in terms of manufacturers, scope of applications, specifications and economic factors. This section will be a comparison for the wide range of cameras available with mentioning relation with the project.

As been discussed in previous sections, many factors and features are found in every camera. So, the number of products which apply these features is big. List of features which describe the diversity of products includes:

- Resolution
- Frame Rate
- Sensor Type: CMOS or CCD
- Color of output: Monochrome, color or other
- Price
- Bits-per-pixel: A-to-D conversion
- Compatibility
- And sometimes the optics used

Depending on the application, the choice is different. Video cameras range from an internet webcam to a sophisticated scientific one. As the project tries to solve an industrial problem, choices are limited to some camera systems. Most of industrial cameras are FireWire cameras. In the same time, the domain of FireWire Cameras is big and depends also on the factors mentioned above. Here is a table which brings several camera systems:

Table 3-1 List of Different Cameras

Brand	Resolution	FPS	Sensor Type	Bpp	Color	Price
Apple iSight	640x480	30	CCD	8	color	€145
Basler A601f	656x491	60	CMOS	10	mono	€999
Basler A602f	656x491	100	CMOS	10	mono	\$1475
Canon EOS-1Ds-MKII	4992x3328	4	CMOS	12	color	\$7000

C-Cam BCi4-6600	2208x3000	5	CMOS	12	mono	€1920
Dagexcel V	640 x 480	75	CCD	8	mono	\$2000
DBK 21AF04-Z	640 x 480	60	CCD	8	color	€660
DragonFly Express	640 x 480	200	CCD	8	color	\$1100
Fuji FinePix S2 Pro	3024x2016	2	CCD	12	color	\$2400
Hitachi KP-F83F	1037 x 779	30	CCD	10	mono	\$1275
iRez 1394 KD	640 x 480	30	CCD	8	color	\$150
Kappa DX1(0)	640 x 480	30	CCD	10	mono	\$890
Leaf Valeo 22wi	5356x4056	0.8	-	16	color	\$27000
LW-SLIS-2048A	2048 x 1	30000	CMOS	12	mono	\$1700
NETGmBH FO1224TB	752 x 480	60	CMOS	12	mono	€495
NETGmBH FO234SC	782 x 582	63	CCD	12	color	€1090
PCO 1200 HS	1280x1024	625	CMOS	10	mono	€22500
Phantom v4.1	512 x 512	1000	CMOS	8	mono	\$28000
Phantom v9	1632x1200	1019	CMOS	8	color	\$84000
Phase One P21	4904x3678	1	CCD	16	color	€15000
Photron Ultima 1024	1024x1024	500	CMOS	-	color	€38500
Phytec CAM-111H	1024 x 768	30	CCD	8	mono	€740
Point Grey Scorpion	640 x 480	60	CCD	12	mono	\$1195
Prosilica CV640	659 x 493	120	CMOS	10	color	\$2100
QuadHDTV	3840x2160	30	-	12	color	\$46000

Scion CFW-1312	1360x1024	7.5	CCD	12	color	\$2495
Sony XCD-SX910UV	1280x1024	7.5	CCD	8	UV	\$6595
Sony XCD-V50	640 x 480	60	CCD	8	mono	\$1520
Unibrain Fire-i	640 x 480	30	CCD	8	color	\$90

Source: [Douxchamp], see References

Bpp = “Bits-per-pixels”

FPS = “Frame per Second”

UV = “Ultraviolet”

For bigger list of products, go to:

<http://damien.douxchamps.net/ieee1394/cameras/index.php>



Figure 3.5 Camera Example from Allied Vision Technologies



Figure 3.6 Camera Example from Basler

3.1.1.3 Frame Rate Analysis

The process of capturing images of a moving object is ultimately a sampling process. From the continuous change in the infinite positions the fan assumes as it rotates, discrete snapshots –or samples– are taken by a camera. The sampling is done at discrete and specified time intervals, hence there is a *sampling frequency* inversely proportional to this time period, and Nyquist sampling rate must be considered.

This section addresses the issue of sampling rate requirements for the two image processing techniques used in this project, namely mark extraction and edge detection. The analysis results here are vital in determining the

algorithm with lowest cost in the implementation phase, since frame rates of cameras significantly affect their cost.

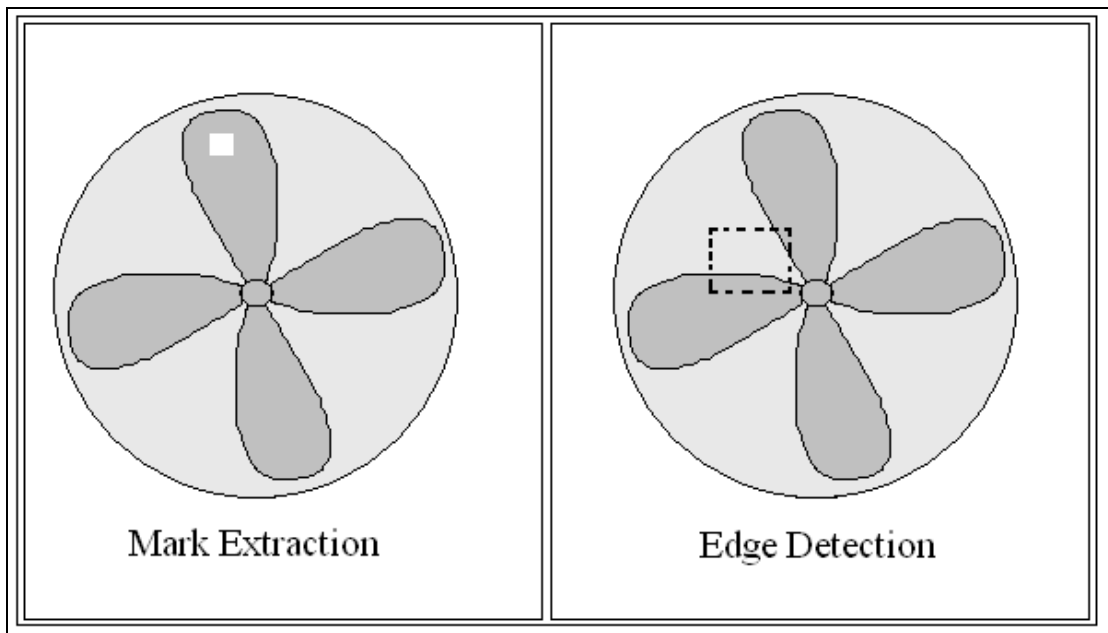


Figure 3.7 Frame Rate Analysis

3.1.1.3a Frame Rate for Mark Extraction

Assuming a full view of the image is used in the processing of the mark extraction algorithm, the frequency of the mark repeating a revolution is obviously equal to the frequency of the fan. ***The number of blades in this situation is irrelevant.*** Hence the problem is analogous to taking sample readings of the minutes hand of a clock, where the frequency should be higher than 2 readings per hour to avoid aliasing since the frequency related to completing a revolution is 1 revolution of the minutes hand per hour.

Thus if the frequency of the fan (the revolutions per minute) were 900 RPM in reality, a sampling rate (frame rate) of over

$$2 \times \frac{900}{60} = 30 \text{ fps}$$

is required.

This lead to the **following result**:

Given that the industrial fans in the ACs must be in a range between 800 and 2000 rpm, the camera used should have a sampling rate able to measure over 2000 rpm to detect a fault due to high speed, assume 2200 rpm is sufficient, then the frame rate in seconds of the camera required should be no less than:

$$2 \times \frac{2200}{60} = 73.33 \text{ fps}$$

This number is a must for worst case scenarios, even though normally 40 fps would suffice if the fans did not usually increase above 1200 RPM.

3.1.1.3b Frame Rate for Edge Detection

In this case, it is assumed that *all the blades are indistinguishable from each other by any characteristic*. The problem here is bigger since given any two random images of the fan in a specific revolution does not enable the observer to determine the angle difference between the two, in contrary to the case where a distinguishing mark is present.

Hence in this situation, information can only be obtained if sampling is higher than Nyquist rate relevant to blade crossing frequency and the algorithm keeps track of the passing blades. The number of blades here is essential in determining required fps.

The formula here for computing the required fps given the max. rpm is modified to:

$$fps \geq \frac{2 \cdot N \cdot RPM_{\max}}{60}$$

Where N is the number of blades of the fan. The simple computation was left undone to show the explicit elements in effect. This leads to the **following result:**

Given the same max, RPM assumption in the previous case, and the fact that the industrial fan has 4 blades (which was not relevant in the previous case), the frame rate of the camera should be no less than

$$2 \times 4 \times \frac{2200}{60} = 293.33 \text{ fps}$$

Hence a crucial fact is in order. The frame rate in the case of edge detection is higher than that of mark extraction by a multiple of N , namely, by the number of blades. However, the cost of a camera increases rapidly with the increase of frame rate, signaling a severe financial constraint on the method used on edge detection.

3.1.1.4 Project's Camera

This section is discussing the work of the project which is related to the decision of camera choice to set it as the camera to be used through the course of the project. The choice has been altered many times before finalizing. This was due to many administrative, economic and time factors, which all will be discussed in the following paragraphs. Also, experiments done for the selected camera will be reviewed.

Project's tentative constraints for camera choice are:

- Relatively *high* frame rate to be able capture the motion of the fan
- Financial issues are governed by an outside-university (third-party) entity
- Technical aspects are restricted by the available computer software and hardware in campus of King Fahd University.

As requested, getting a camera in the beginning of project period was not critical as it is not the implementation phase. Project work tries to design the

proposed system and evaluate the end-product before implementing. So, initial design work started with an internet WebCam to work with the software development (which will be discussed in section 3.3 of the report).

The plan was to complete software development to the point before engaging project's fan. Unfortunately, the outside (third-party) funding entity expressed extra demands before providing the support of a higher speed camera. So, the choice has been altered *intelligently* to work with the available resources in order to complete the project with the set objectives. The decision was to keep on working with the WebCam to measure smaller speeds of the fan, as will be explained thoroughly in section 3.1.2.

So, a USB WebCam was used. It is Genius® Slim 320. Slim 320 specifications are:

- Image Sensor: CMOS
- Image Size: 640 x 480 pixels (for video)
- Frame Rate: up to 30 fps (further comments in following paragraphs)
- Interface: USB 2.0 ~ 400 Mbit-per-second
- Geometric Dimensions: 38.8 x 33.8 x 92 mm (Genius®)



Figure 3.8 Slim 320 in Lab.

As shown above, Slim 320 was a suitable camera to use. Engaging and usage of the camera will also be discussed through the Software Development section of the report (3.3).

Regarding the comment beside the frame rate of Slim 320, the '30 FPS' speed is controversial! The manufacturer, Genius®, claims this speed, hence it is not. This speed can be reached at optimal conditions, ambient or device conditions. As explained in section 3.1.1.1, [3) Frame Rate], the '30 FPS' number is a combination of two speeds of the camera, namely, exposure period and digitizing and transferring time. So, '30 FPS' does not imply necessarily that the camera will capture 30 images every 1 second passed

for the view in front of the camera, i.e. each image corresponds to ~ 33 ms. So, an experiment has been done to investigate the actual capturing capabilities of the camera.

The experiment has been conducted using the support of an outside time source to get the actual pseudo-speed of the camera. Two ways has been done. First way is to capture an enormous number of frames of the movement of a clock hands for one minute (a naïve method!). This first experiment resulted in an approximate frame rate of 25-28 frames per second. The second way of experiment conducted utilized the use of software to do the outside time source. As ‘How?’ will be explained in Software Development section, the experiment simply is based on collecting time stamp of each frame coming to the computer. This method resulted in more justified numbers. The Slim 320 camera tends to have fluctuated number of frames per second for the beginning of establishing the capturing mode. However, the camera then stabilizes its frame rate (almost) to about 20 frames per second. These collection of experiments resulted in valuable figures before beginning the development of software for computing the speed of the fan.

3.1.2 Fan

3.1.2.1 Search Process

The search for an appropriate fan started by inspecting the AC provided to the Systems Engineering Senior Projects Laboratory from the Mechanical Engineering Department. The AC was, however, well sealed and there was hardly any visual –let alone physical– access to the fan.

After seeking help from the project advisor, Dr. El-Ferik, the team was provided an old office metallic fan. The structure was bulky, and removing the case was also quit dangerous due to the metallic composition of the blades. Even so, this fan was initially used, and progress was made with the initial mark-detection algorithm developed. Refer to 3.3.2.1.

The team favored working with a more practical fan, however, one that had plastic blades and was smaller for easier transporting and positioning. In effect, a fan of such characteristics was purchased and disassembled. The blades had a light gray color and were painted black to match the real industrial fans and, in addition, increase the reliability of edge detection (since a lot of inaccuracy was introduced into the images after necessarily being converted to 8-bit images in the software).

Until that period, the team was only concerned with capturing the images and designing the basic back-bone algorithm that would detect the mark or edge of a fan from an image, with total disregard to rotation and movement at that phase. It took a while to understand and further implement the simple task of acquiring an image in LabVIEW, converting it to an 8-bit image, and then processing it to extract information, before worrying about measuring speed from image sequences.

What's more, the team was expecting a high speed camera in the near future once they started capturing image sequences (videos) of the fan in motion.

The problem with all previous fans emerged when the team had developed the basic algorithm to measure the fan speed. Consistent reference speeds were to be known in advance in order to test the validity of the software. However, the high speed camera was not provided as mentioned in the previous section, and there was no reliable instrument in the teams possession yet to measure the speed anyway.

The team was instructed by Dr. El-Ferik to slow down the fan so that the camera they had (described in the previous section) would be able to capture the blades without much blur. As images were captured of the fan at different low speeds, the team noticed that the camera could not handle speeds over 40 rpm without significant blur. This number was computed by simply counting the number of rotations of a marked blade in one minute and repeating the process for a rough average estimate.

Hence the next step towards finding reference speeds of the fan required slowing it down to very low, yet steady and consistent, speeds. After a few discussions with the lab engineer, Mr. Shaukat, it was clear to the team that it would be wiser to merely *decrease the input voltage to the motor instead of interfering with its internal circuitry.*

As a result, a continuous voltage transformer was purchased. It received an input of 110V and the output could be varied in a range between 0 and 110 volts. This succeeded in reducing the speed of the fan considerably, but the continuous speed reduction had a limit. The fan would decrease its speed continuously as the voltage was stepped down from 110 to 50V, but after this threshold, the fan would completely stop, refusing –as it were– to apply any torque no matter how small to move the blades. The speed of the fan given a 50 volt input was still too fast for the camera to capture.

The team was informed by Dr. El-Ferik that the motor in hand was an induction motor that had a threshold operating voltage, which had nothing to do with overcoming inertia. It simply didn't assist to give the motor a voltage impulse or mechanical startup force to start, and then regulate it to a voltage input below 50 volts.

To overcome this problem, the team thought of either reducing the number of turns in the motor coil, or simply introducing a source of friction or inertia. The latter method, although naïve, was undertaken first since it offered no potential risk of completely damaging the motor as did the former.

Consequently, heavy objects were attached to the blades in hope of reducing its speed at 50 volts. Duct tape was also rapped around the outer metallic surface of the motor, by that mechanically resisting the motion of the blades. This, however, introduced viscous, not linear, friction. One extra layer of tape completely stopped the fan, while removing it returned the fan to its high speed rotation once again.

But even if that worked, it nevertheless would not solve the problem of finding a reliable set of reference speeds of the fan at different voltage inputs, which was necessary to test the outcome of the project. This fact was very disturbing to the team.

3.1.2.2 Selection and Solution

After a lot of frustration, the team thought of a great idea upon recalling the work in previous projects: why not use a small DC motor? After all, several labs had an abundance of modern motors that had high torque and a very accurate linear voltage-rpm mapping. Several motors were found, and the team selected a DC motor with a 0-12 voltage input providing a proportional output ranging between 0 and 80 rpm.

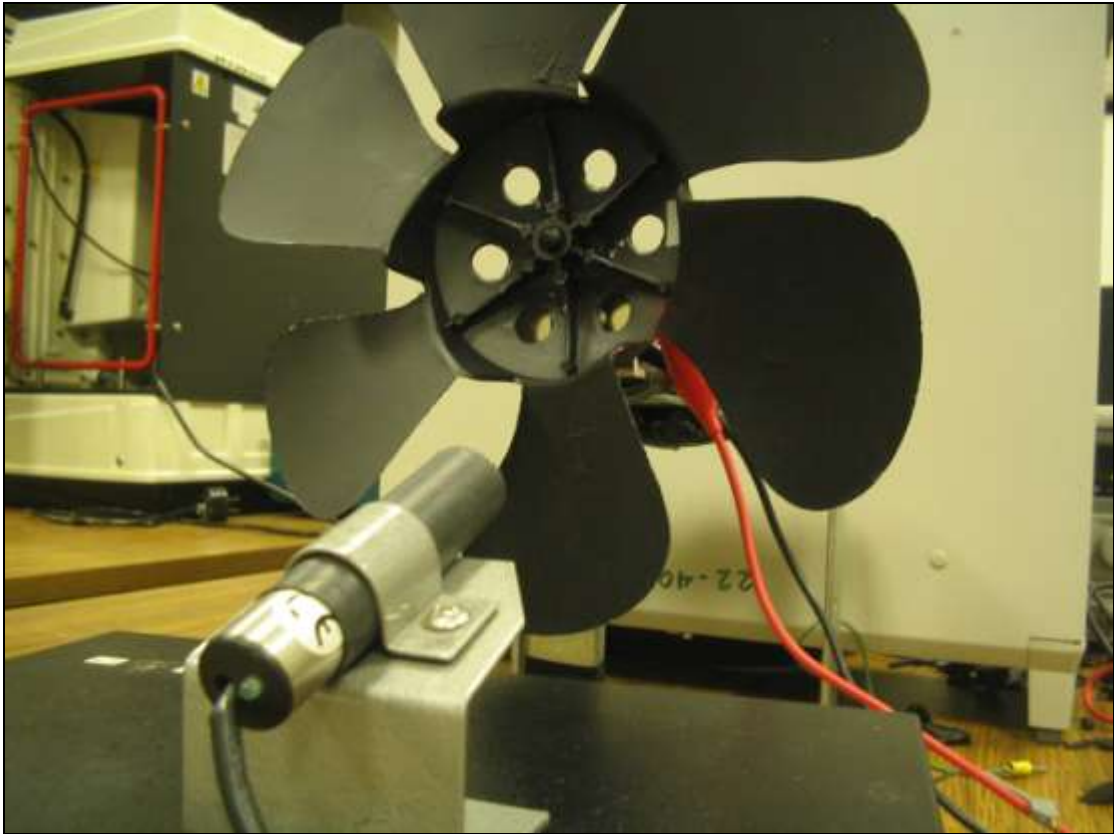


Figure 3.9 Fan Selected

Moreover, the optical tachometer provided by the Electrical Engineering Department –which was then recently provided to the team– confirmed that this output range was not affected by the load, namely the blades, attached to the motor.

As the team at this level had the optical tachometer along with the DC motor in hand, it was finally possible to have solid –even continuous– reference of input voltages vs. fan speeds to check the validity of the projects outcomes at a wide range of fan speeds. This signaled the completion of a very important phase in the project.

3.1.3 Photo Tachometer

In the design of any new measurement system, the output must be tested by an independent reference. The output measurement of the designed system must be heavily correlated with the reference output to assume proper functioning of the measurement system.

Since the measurement in this project does not consider dynamic response of the sensor, nor the fluctuation of the sensed variable in time (one average speed is displayed for each fan, since it is assumed the rotational speed at this phase is roughly constant), the most reliable sensing scheme found in this case was the reflexive optical tachometer. (as stated earlier, some scientific sources confine the term ‘tachometer’ to instantaneous speed sensors, in which this optic sensor would not be labeled as a tachometer.)

The team was provided such a sensor from the Power Systems Lab of the Electrical Engineering Department: a Hampden Digital Photo Tachometer, model HPT 100A. The sensor’s technology is based on laser spectroscopy. Basically, while a laser source constantly emits a laser beam, a small sticker – serving as a mirror– placed on the rotating object reflects the laser beam as it periodically crosses its projection on the rotating object.



Figure 3.10 Hampden HPT-100A in operation

The measurement range of this sensor is stated to be between 50 to 10,000 rpm in the datasheet. However, it proved to give accurate values even for lower speeds reaching 5 rpm. This speed is slow enough to measure by the naked eye so that confirming accuracy at these speeds was not a problem.

The resolution is 1 rpm. Even though this is more than enough for the purpose the sensor was manufactured for, it posed a slight problem for us when dealing with very low speeds where half an rpm is relatively significant. This was not very critical, though, even in our case.

3.1.4 Desktop Computer

This very brief section is about the PC used to operate all the software and hardware of the project. Generally, the PC which will be operating has the normal of specification of any PC. The main issue is the issue of compatibility with purchased hardware and software. For example, a FireWire camera needs FireWire port to be in the PC, of course. The software (which will be discussed in later sections of the report) is dependable upon the operating systems of the PC, but in most of the time the software used is compatible with existing operating systems.

During the work of the project, the design process took place in '*Industrial Automation Lab*' of Systems Engineering Department, Room 411, Building 22 at the King Fahd University of Petroleum and Minerals campus. The PC used is manufactured by HP Compaq. The specifications of the PC are summarized into:

- Intel® Microprocessor
- Pentium 4, 3.20 GHz
- RAM 512MB
- Windows XP Service Pack 2
- USB 2.0

3.1.5 Other Equipment

3.1.5.1 Continuous Voltage Transformer

This device was used in an early phase of the project development when the high speed camera was not provided and it was required to slow down the fan to suite the webcam used in the project.

this device receives an AC input voltage of 110V and transforms it (continuously) to lower AC voltage levels ranging from 110 to 0 volts.



Figure 3.11 Continuous Transformer



Figure 3.11 Continuous Transformer from back

3.1.5.2 Metal Mounting

Four aluminum L-shaped metals were purchased to provide structure for the new simple assembled fan (composed of the DC motor and the blades). They were connected in a way that would insure stability and ease of positioning of the fan.

3.1.5.3 Trip Switches

These force sensors were used in an early phase of the project development before the team was offered the optical tachometer from the Electrical Engineering Department.

Although inaccurate and naïve, the method was still usable. Rubbers would be attached to the trip switches and slightly exposed to the blades as they

rotated which provided the switching force. The output voltage function would be fed to the computer where a simple algorithm in LabVIEW could be used to determine the velocity.

Others: although not of direct relevance to the concepts of the design process, some devices and instruments were vital in completing the project. These include a digital voltage source and a digital multimeter.

3.2 Image Processing Background

Before discussing software development of the project, it is essential to clarify the theoretical concepts behind the vision system solution of the project. In more general words, this section will discuss the concepts of Image Processing. Also, Mathematical foundations of the field will be discussed. Special techniques related to the project will be discussed in more explanation. Some modern applications will also be reviewed.

3.2.1 Digital Image Processing: Introduction

In this section, brief introduction to concepts of Digital Image Processing will be reviewed. The word ‘digital’ correspond to the application of computer technology. Digital Image Processing is the field of applying mathematical algorithms through computer programming onto digital images. The digital image is represented as a 2D array or matrix of pixels. Each pixel holds a numerical value of the gray level for grayscale images, or values of RGB (Red, Green, and Blue) combination.

The whole image is a sampled data of the actual view captured by the camera (see section 3.1). The sampling process depends on *quantization levels* of the camera system standard. Most of Digital Images are digitized onto 8bit form. This form means that for a grayscale image, each pixel carries one byte of gray level data; and for color image, each pixel carries 3 bytes for each color data. The 8bit form contains 256 gray level, making 0 as black and 255 as white. The same thing is for RGB for each color (i.e. 256 x 256 x

256 possible combinations). Color level or gray level also is called **intensity** as for light intensity.

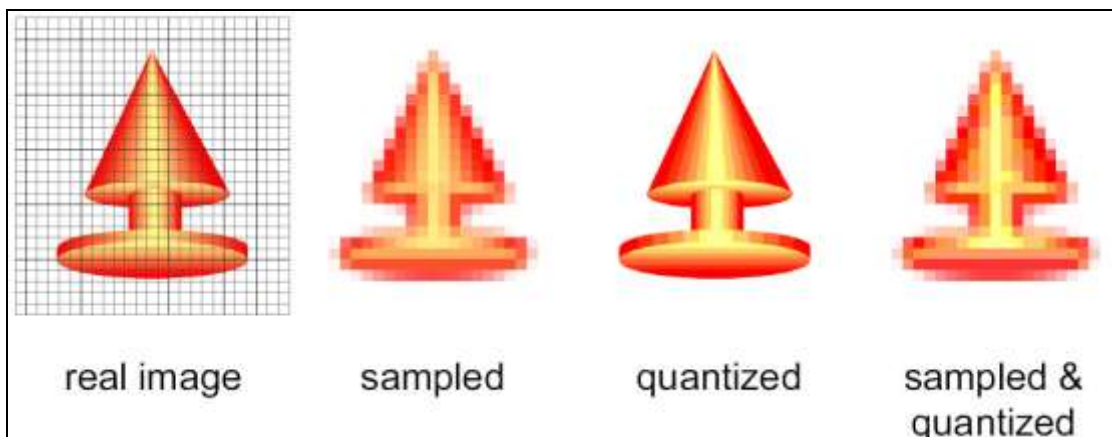


Figure 3.12 Steps of Digitizing an Image

[Source: Richard Alan Peters II, (2007), Vanderbilt University]

For most digital image processing applications, monochrome (grayscale) images are used. That is because the ease of dealing with smaller amount of data with no loss in image information unless color information is required specifically. Mathematical approach of image processing splits into many disciplines. Generally, the term “Image Processing” also includes, of course, both the concepts of processing and analysis.

There are different levels of kinds of processing an image:

- *Low-level processes*: contrast manipulation, brightness adjustment, and all other image editing functions (e.g. Photoshop)
- *Medium-level processes*: segmentation, recognition, etc.
- *High-level processes*: objects groups understanding

Normal Steps of Image Processing:

- 1 - Acquisition: (talked about in 3.1)
- 2 - Pre-processing: image enhancement, noise reduction, etc.
- 3 - Segmentation:
- 4 - Representation and description:
- 5 - Recognition:
- 6 - Knowledge Base:

Steps from 1 to 2 are basis for different of applications which their only aim is to display the image for further human usage, i.e. without the concept of automation. Going steps 3 through 6, image processing transfers into area of information understanding. Further advancements raise the complexity of the system, especially when approaching the related fields of artificial intelligence and robotics.

- Image Array Manipulation

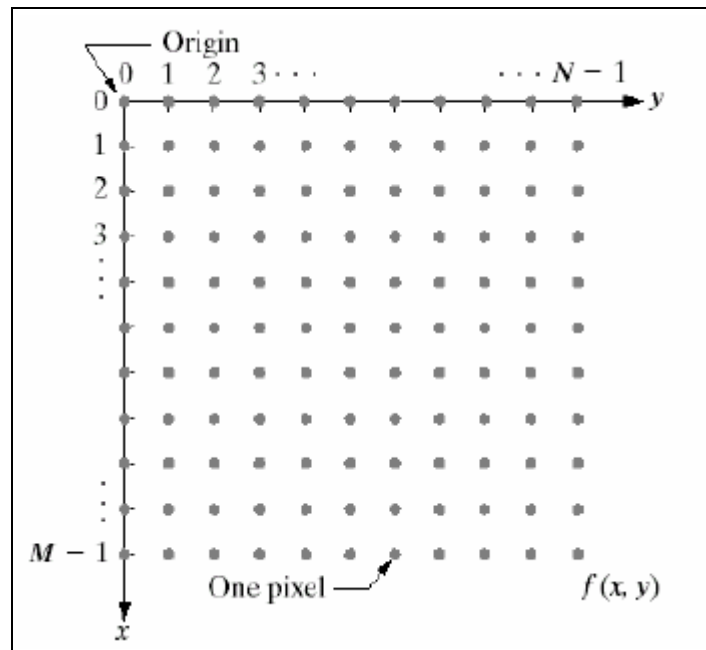


Figure 3.13 Image Matrix

[Source: Gonzalez and Woods, (2002), Digital Image Processing]

As shown in figure 3.14, image matrix (array) are represented as any mathematical matrix (observe the origin). With dimensions of the matrix correspond to dimensions of the image (of course!), each matrix element carries its gray level value ranging from 0 (black) to ∞ (white) or simply 255 for an 8bit image.

There are two basic approaches for image enhancement:

- Spatial Domain techniques
- Frequency Domain techniques

Spatial domain image processing techniques depend on the direct mathematical operation on the values of the pixels of the image.

Spatial domain techniques include:

- Image negatives
- Log transforms
- Power transforms
- Contrast stretching
- Image gray levels Histogram Processing
- Arithmetic or logic operations
- Smoothing filters (also via frequency domain)
- Sharpening filters (also via frequency domain)

Frequency domain techniques are those which apply mathematical operations on the frequency transformation of the image, i.e. Fourier Transform. As “time” here is described as analogy to gray level occurrences throughout the image. Fourier Transform is applied into the two dimensions. So, 2D discrete Fourier transform is used. Moreover, for ease of application, in image processing, Fast Fourier Transform is used rather than the original one to speed up the processing. Techniques of frequency domain enhancements include:

- Smoothing filtering: collection of low-pass filters
- Sharpening filtering: collection of high-pass filters

Noise processing in images has its own branch utilizing both the spatial and frequency techniques.

Other image manipulations are found depending on the application. As image enhancements are fulfilling step 2 of normal image processing, further steps are related for higher level of processing. More details will be given in following sections for the fields of:

- 1) *Feature Extraction*
- 2) *Morphological Image Processing*

These two fields are explained more elaborately because of their usage in our project as it will be discussed in Software Development section (See 3.3). Other fields of Image processing that are also fulfilling steps 3 onwards of a normal image processing:

- Color Image Processing
- Image Data Compression: telecommunications and multimedia
- Object Recognition: Artificial Intelligence
- Pattern Recognition: Artificial Intelligence
- Wavelet Image Processing

3.2.2 Feature Extraction

Feature Extraction section will be dealing with information-approach of the image data processed. Feature extraction concept is used generally for any sort of data. The aim of the extraction is to separate a small amount of information which describe the whole set of original data accurately.

In this section of the report is set for discussing feature extraction concept in digital images. Review of methods will be discussed. Special method or feature will be discussed in more detail, which is **Edge Detection**. That is because that Edge Detection is the most important tool used for the solution of the project among other tools also used.

3.2.2.1 Methods

Many methods are used in feature extraction from a digital image. All depend on several mathematical concepts. All of the following methods are applied to grayscale images operating in the gray levels or simply intensity. List of features which have methods to extract them in *image processing literature*:

- Corner Detection
- Interest Point Detection: is more general method than corner detection. This also includes with corner detection, isolated point with local intensity extrema, end of lines, or others.

- **Blob Detection:** is based on looking for points (or regions) with more (or less) brightness than surroundings. Applications include: object recognition, object tracking and texture recognition
- **Ridge Detection:** (or Valley Detection) is a way of extracting the information of smooth two-variable function (i.e. image) looking for maxima or minima, i.e. valleys. This method is not related to maxima or minima in the intensity levels. It is more related to a graphical approach. Applications include object recognition.
- **Edge Direction**
- **Motion Detection:** is the process of sensing physical movement. Specifically in images, a simple algorithm is to compare pixel values with a reference image's values. Applications include security and other.
- **Optical Flow:** This is used in the estimation of motion. From the name "flow", it is related to vector field analysis of image data.

Higher-level extraction techniques include:

- **Thresholding:** it is a very large field in image segmentation. It is basically the process of replacing each pixel intensity level with either a "HIGH", "1" value or "LOW", "0" value; depending on a specific thresholding criteria. All of the work is on the criteria. Methods include: basic global thresholding, basic adaptive thresholding, optimal thresholding, local thresholding or even Entropy-based (information theory) algorithms. Hence the output of a thresholding

process is a binary image, i.e. two levels, normally black, 0; and white, 255.

- **Blob Extraction:** is related to the categorization of discrete regions in a binary image.
- **Hough Transform:** this transform is initially developed to be used in finding lines in images. The technique is then extended to identify other shapes like circles and ellipses. Generalized transformations are developed also to identify other arbitrary shapes given their models.

3.2.2.2 Edge Detection

In this section, a special feature extraction methodology is discussed in more detail. That is because it is used as the main technique in the project. Here, edge detection will be reviewed from the theoretical foundations of the technique. The usage of the technique will be shown in detail in the software development section.

As been explained in beginning of discussion of feature extraction, one of most important field of research is edge detection. In an image, an edge is defined as a set of connected pixels that are on the boundary between two regions. This definition is general. In more specific meaning, *ideal edge* is the set of connected pixels located at step transition in gray level.

Normally, edges in digital images are not 'ideal'. Most of the time, blurring is associated with the edges. This blurring is due to:

- Quality of the acquisition system (camera),
- Frame rate, or
- Ambient illumination

So, usually, edges occur as a ramp transition in gray level. Ramp slope is related to the degree of blurring. The **1st derivative** of the gray level values of line profile of the ramp transition would generate a constant nonzero value as we go over the transition portion. Other constant gray level values elsewhere will have a zero 1st derivative. Normally, 1st derivative nonzero value is positive for a dark-to-light (0-to-255) transition. **2nd derivative** of the line profile would generate a positive value for beginning of transition at the 'dark' side and a negative value at the end of the transition at the 'light' side. The 2nd derivative also gives the information of a 'zero-crossing' in the middle of the transition area. See figure 3.11 for more explanation of the concept.

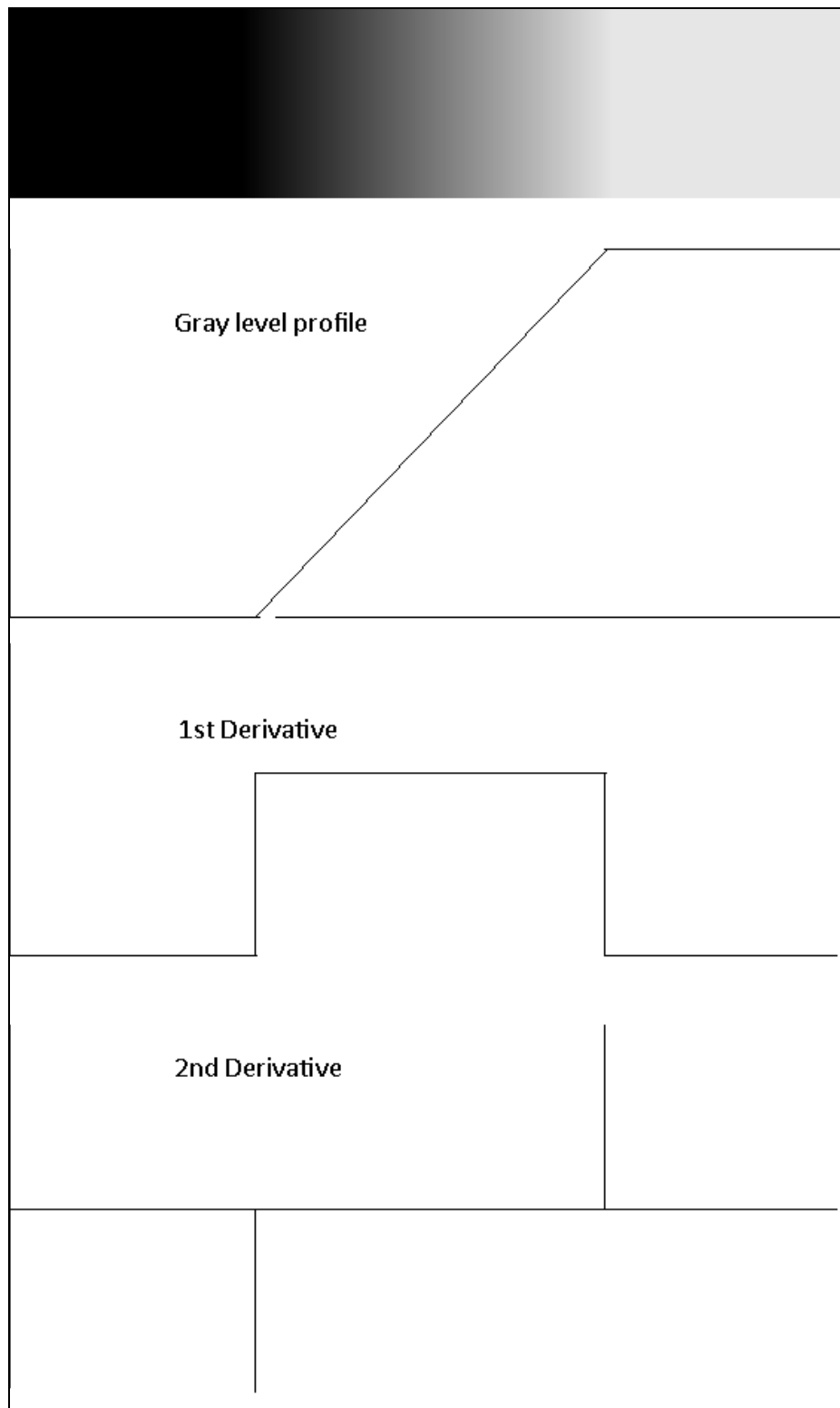


Figure 3.14 Edge Mathematical Analysis

Although edge analysis up till now is only about for a 1D line profile, the same principle is used for any edge in an image. As the image is represented as a 2D matrix, 1st derivative is based on 2D gradient of the image function $f(x, y)$, i.e. ∇f . 2nd derivative of the image matrix is based on the Laplacian of the 2D function, i.e. $\nabla^2 f$.

In 2D images, edge detection algorithms depend on application of gradient operators. That is because the computation of ∇f for a discrete pixel values requires using special operators. Usually, these operators are 3 x 3 ‘masks’ that are applied to the whole image matrix through **Convolution** to give an approximate gradient result.

1st order operators include:

- Canny edge detector: is a multi-stage algorithm. The algorithm parameters allow flexibility which lead to better results depending on the particular situation.
- Sobel operator: give some inaccurate approximation of image gradient, but it is useful in applications.
- Prewitt operator: has the same properties of Sobel; and is one of the most used in practice.
- Roberts’ Cross operator: is one of the earliest algorithms. It uses a simple 2x2 mask. It is more noise sensitive and inaccurate, but faster in computation.

2nd order operators include:

- Marr-Hildreth algorithm: utilizes the 2nd derivative property through the detection of the zero-crossing. However, this algorithm returns some false edge.
- Zero-Crossing or Laplacian filter: is, as from the name, use the principle of searching for a zero-crossing in the 2nd derivative of the image.

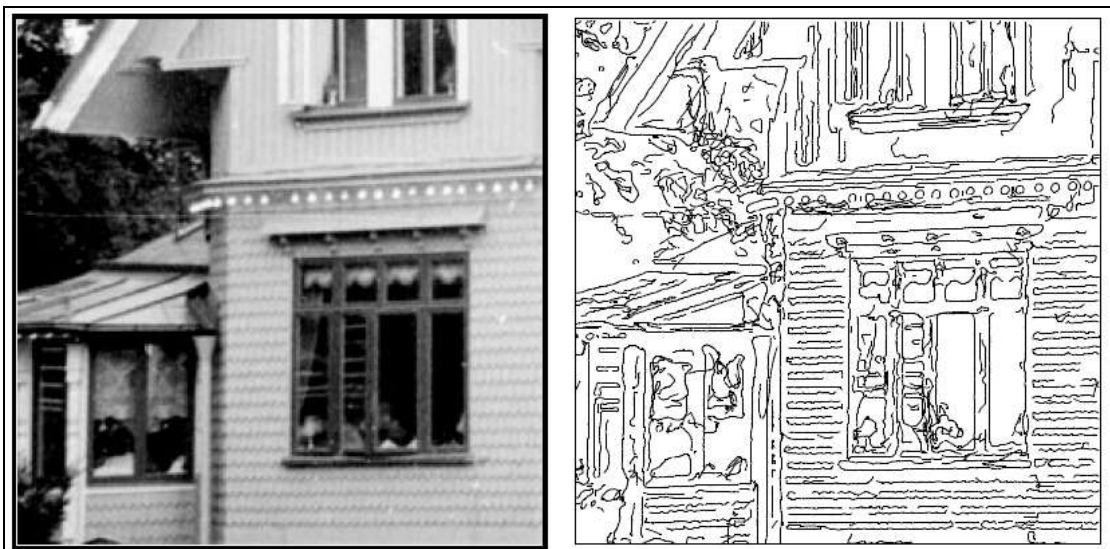


Figure 3.15 Example of Full Image Edge Detection

[Source: Tony Lindeberg, (1998), Computational Vision and Active Perception Laboratory, Royal Institute of Technology, Stockholm, Sweden]

Edge detection proved that it is a powerful tool of image segmentation and feature extraction. So, an enormous number of applications is expected to have a phase of edge detection in it. One application is the usage of the technique in this project!

3.2.3 Morphological Image Processing

In this minor section of the report, another field of image processing is discussed. The reason is that this concept is used sometime among the project work. It is used in an attempted algorithm in the software development but rather rejected for the edge detection solution. More details of the algorithm are found in section 3.3.2.1.

A big field of image processing which depends only on binary image data is called morphological image processing. This concept is taking its principles from the field of **Mathematical Morphology**. Mathematical morphology depends on **Set Theory**. These techniques are useful for description and representation of regions in an image. Furthermore, other purposes are done through morphological filtering. On the contrary of other image processing techniques, morphological techniques handle and operate on the organization and order of pixel values rather than the numerical values themselves. Sometimes, morphological techniques are used on grayscale images, but are limited.

‘Sets’ in binary images represent objects and regions in the image. So, all of techniques under morphology are using set theory operations of \cap , \cup , etc. Logic operations also applicable like NOT, AND, XOR, etc.

Here are a collection of morphological operations:

- Dilation: the dilation of A by B results on the set of all displacements that B could do while overlapping between A and B is found even for one element. Set B is often called a structuring element.
- Erosion: erosion of A by B results in set of all displacements that B is wholly contained by A.

Both the morphological operations of dilation and erosion are the basis for most of other morphological operations.

- Opening: as it is clear from name, opening operation tries to remove small objects of the image and tries to smooth the contours of objects. Opening is also used to find specific shapes in the image. Opening is a combination of both dilation and erosion operations.
- Closing: as opposing to opening, closing tries to eliminate small holes and fill in gaps. It is also, of course, based on combination of both erosion and dilation operations.
- Hit-or-Miss transformation: is a basic tool for shape detection. The result of this operation is the position of the desired shape being the structuring element.
- Boundary Extraction: is performed by firstly getting the erosion of A by B and then doing the set subtraction between A and the erosion result. A is the shape set that its boundary to extracted and B is a suitable structuring element.
- Region Filling: is done using dilations, complementing and intersections operations to get the desired shape to be filled.

- Convex Hull: for a set A to be convex, if any line joining any two points in A is inside A. So, a convex hull of a specific set is the smallest convex set to contain this specific set.
- Thinning: a normal thinning operation is to sequentially apply thinning A by a structuring element B to have at the end a thinned shape.
- Thickening: is the dual operation of thinning.
- Skeletonization: as clear from name, this technique is aiming for getting the skeleton of a specific shape. Finding the skeleton of a set A is defined by combination of erosion and opening operations.
- Pruning: is a method of cleaning up the results of thinning and skeletonization.

A similar approach is done for morphological processing for grayscale images. Some applications of grayscale morphology, other the basic ones, include:

- Morphological Smoothing
- Morphological gradient
- Textural segmentation
- Granulometry: field of determining size distribution of particles in an image.

As been shown in this section, morphological image processing deals more deeply with information and ‘understanding’ concept of images. As it is considered somewhat advanced methods, the basis is only depending on the Set Theory of Mathematics.

3.2.4 Computer Vision and Machine Vision

In this section, more technology approach of image processing is discussed rather than the mathematical background of it. Here, discussion will be tackling recent technology challenges related to the science of Digital Image Processing.

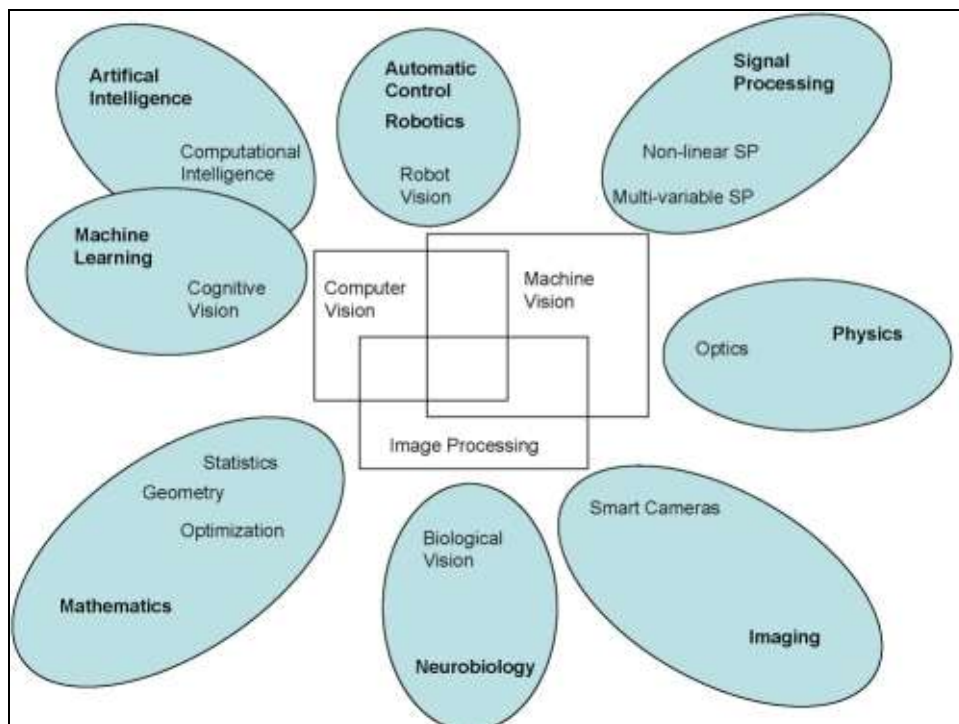


Figure 3.16 Image & Vision related Research Topics Tree

The term ‘Computer Vision’ is defined as science and technology related to computers that ‘see’. The field deals with artificial systems. The term ‘Computer vision’ applications could be:

- Robotic vision
- Autonomous vehicles
- Security Surveillance
- Medical analysis
- Computer-Human interaction

Artificial Intelligence has big role in the development of computer vision. The aspect of automation is the driving wheel of the relation of A.I. and Vision, specially the fields of Pattern Recognition and Machine Learning.

The term ‘**Machine Vision**’ is normally used for the application in industry and manufacturing. In industry, typical application could be a barcode reading, character recognition or visual inspection. This project, as been told in the beginning of the report, applies the technology in a somewhat new application of visual rotation measurement.

Finally, before going to next section of Software Development, a small comment is clarified here. It is been noticed that in *Literature and Research*, all the terms: ‘Computer Vision’, ‘Machine Vision’, ‘Robot Vision’, ‘Image Processing’ and ‘Image Analysis’ are all relatively talk about the same techniques and developments. Although in some research groups or companies could choose the term which best describe their application.

Generally, **Computer Vision** is related to 3D perception and graphics with the relation to Stereo Vision. **Image Processing and Analysis** often is related to application of 2D images with pixel operations. **Machine Vision**, as been mentioned previously, is more used in the industrial applications.

[Note: most of technical information reviewed in Section and subsections of 3.2 is referenced to the Book “Digital Image Processing” by Rafael C. Gonzalez and Richard E. Woods.]

3.3 Algorithm and Software Development

Here comes the major part of project's work. In this section, details of the process of software development will be discussed. In the beginning, an overview of the software used is discussed. Summary of attempted algorithms throughout the project work but not implemented will be the next part. The major part of software development will be the section of the implemented algorithm. An improved algorithm also will be discussed furthermore with the improved results. Software Outputs will be the ending of this major section.

3.3.1 Software Used

In this section, only a summarized overview of the software used will be talked about. General introduction to software will shed the light to nature of programming and application.

3.3.1.1 LabVIEW

LabVIEW is a product of National Instruments Inc. based in Texas, USA. LabVIEW uses the programming method of blocks. It is a block-based programming tool. It has many applications especially in fields related to Data Acquisition. LabVIEW areas of applications include: academic labs, physics labs and industrial applications (e.g. this project).

In the project, LabVIEW version 8.00 is used. Sometimes, work was done on another PC, other than the main one, where a LabVIEW 8.20 was used. Mainly, both versions do not differ.

Most important issue about software in the project is part of LabVIEW which deals with images. It is a special toolkit that is special for image acquisition and vision. It is called the “**Vision and Motion**” toolkit. This toolkit includes several block functions which are divided into:

- NI-IMAQ: related to the image acquisition module, which is called IMAQ.
- Vision Utilities: related to image management
- Image Processing: related to image processing operations
- Machine Vision: related to image operations for specific industrial applications
- NI-IMAQdx: related to image acquisition for FireWire devices
- NI-IMAQ I/O: related to input/output operations of hardware
- IMAQ-USB: related to image acquisition of USB devices
- NI-MOTION: related to motion toolkit blocks

3.3.1.2 NI Vision Assistant

In this section, small amount of information will be discussed about another assistive product of National Instruments. The software is called ‘Vision Assistant’. The role of this product in the project is significant. Originally, Vision Assistant is an NI product which only used for the purpose of testing and prototyping vision related algorithms. It is not a design tool.

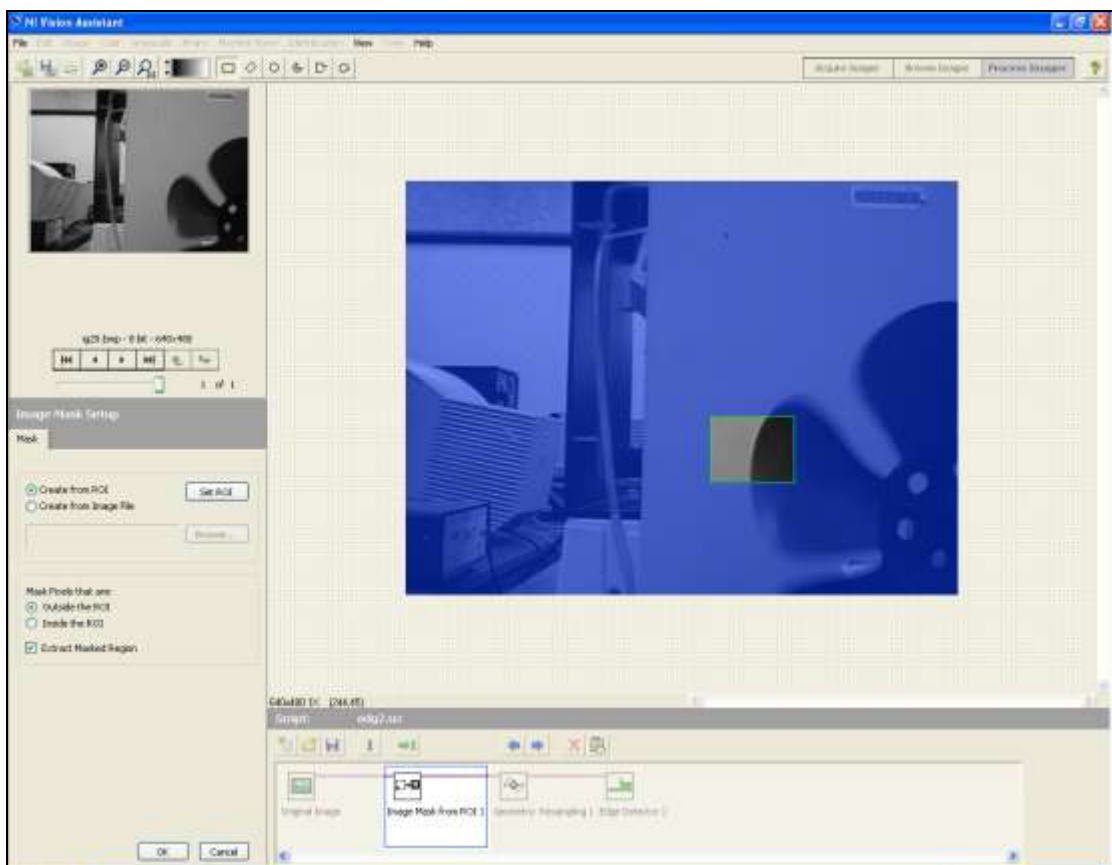


Figure 3.17 Vision Assistant Screenshot

So, for this project, Vision Assistant’s job was to investigate a lot of ways of image manipulation that lead to project’s goals but without any much

programming and debugging. Vision Assistant was used in the phase way before any actual project output was expected. Vision Assistant gave flexibility to try and experiment many useful methods. The overall result of the product is the algorithm that is used in the project which will be explained in the coming parts of the report. Other attempted algorithms is also obtained and tested in Vision Assistant.

3.3.2 Attempted Algorithms

This section describes the logical structure of two separate methods of which one was to be implemented in the project.

3.3.2.1 Mark Extraction

This method was studied first as the appropriate algorithm for the measurement, assuming that it would be feasible to attach a distinguishable mark on one of the fan blades or use some other characteristic of the fan such as a screw. Attaching a mark was, however, more convenient.

The main image processing techniques of this algorithm is based on the principles of mathematical morphology (refer to 3.2.2). The Vision Assistant package of National Instruments was helpful in testing different image processing operations on acquired images.

The logical structure of the algorithm using mark extraction is as follows:

- 1 - Take the image inverse (i.e. its negative)
- 2 - Raise the image pixels to a power of some number.
- 3 - The previous two steps are for the purpose of image enhancement; hence the parameters depend on the features of the image at hand.
- 4 - Thresholding: the method of Metric Auto Threshold was the technique used to convert the image to binary (i.e. black and white).
- 5 - Remove boarder objects. The software utilizes morphological techniques to perform this operation.
- 6 - Edge Detection. This is performed on the entire binary image to recognize the outline of the image objects. (Laplacian)
- 7 - Remove large objects. This depends on morphological properties too.
- 8 - Fill the holes. This also depends on morphological properties.
- 9 - Remove small objects.
- 10 - A block called 'Particle Analysis' is used to give information about the remaining object in the image, i.e. the mark.

This was the last stage reached in software development of the mark extraction method. This was due to the Senior Project Advisor's request of using another technique.

However, assuming the method was selected and used to the end, the previous stages would have been improved in addition to the following essential steps for speed measurement:

- 11 - Given two consecutive images from a video capture, extract the position of the mark in the two images.
- 12 - From the knowledge of the mark positions in the consecutive images and the center of the fan, an angle can be estimated between the three points, giving a measure of the angle change.
- 13 - Given the angle and the time between the two images, the rotational speed can be determined.

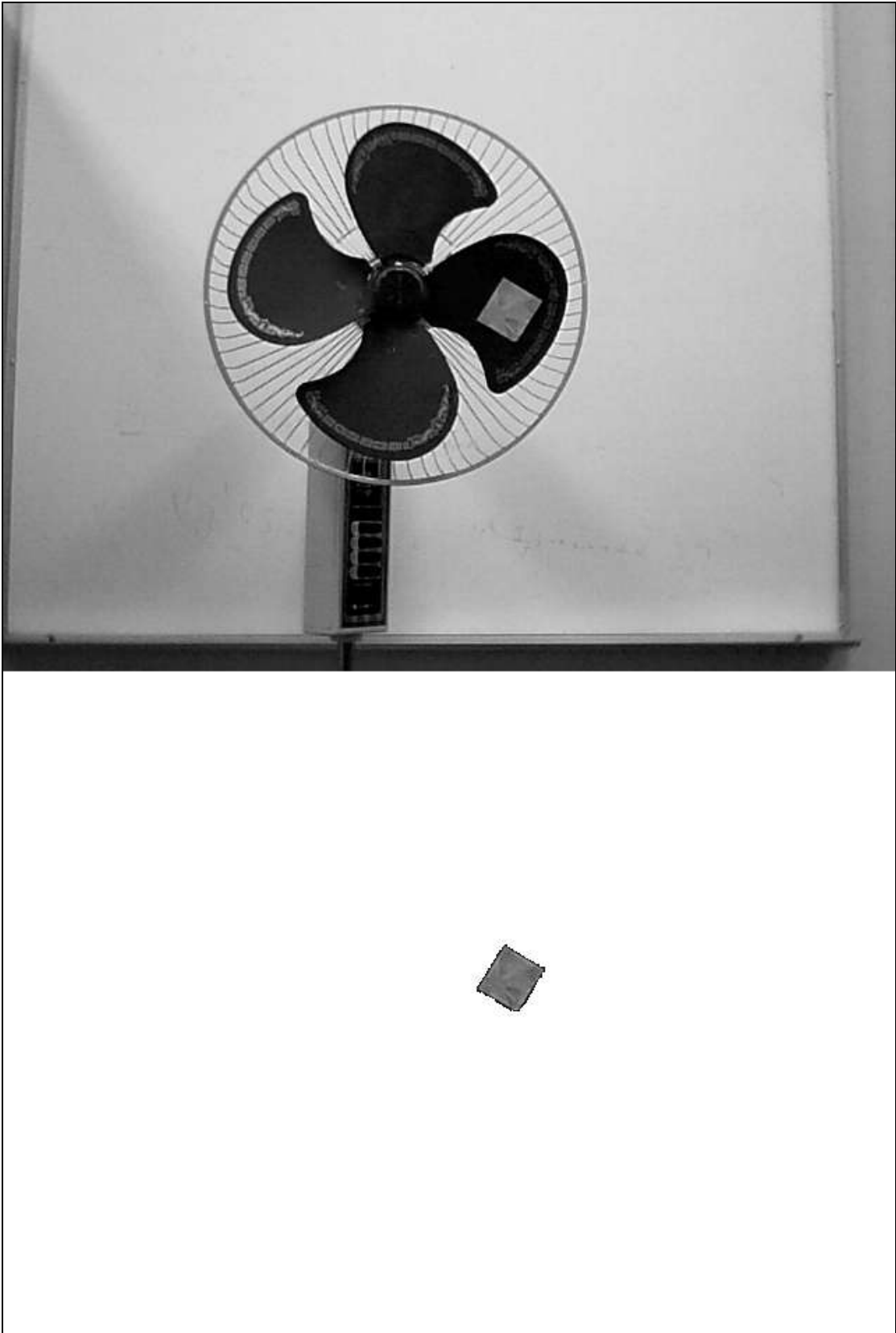


Figure 3.18 Mark Detection. Above: Original Image. Below: Extracted Mark

3.3.2.2 Edge Detection

This was the method requested by the project advisor to be used instead of mark extraction. The complete description of this method is left to the next section.

3.3.3 The Implemented Algorithm

This section contains the core work of the project. It will review in detail the algorithm implemented to calculate the speed of the fan. The algorithm contains seven steps to reach the aimed purpose. This algorithm will be explained in detail arranged as programmed in LabVIEW with illustrations for each step. The following sections are arranged in the same sequence of the algorithm.

3.3.3.1 Step 1: Frame Capturing

Normally, the program starts with performing the acquisition for the images and save temporarily to memory. This step includes:

- 1- Initialization of camera session
- 2- Preparing the image acquisition in LabVIEW for grabbing method
- 3- Start grabbing a preset number of frames using a loop and save them temporarily

4- Snapshots of the captured frames are shown live on the front panel

5- Close the camera session

This step is illustrated in figure 3.20 as capturing 200 frame as default value of the project.

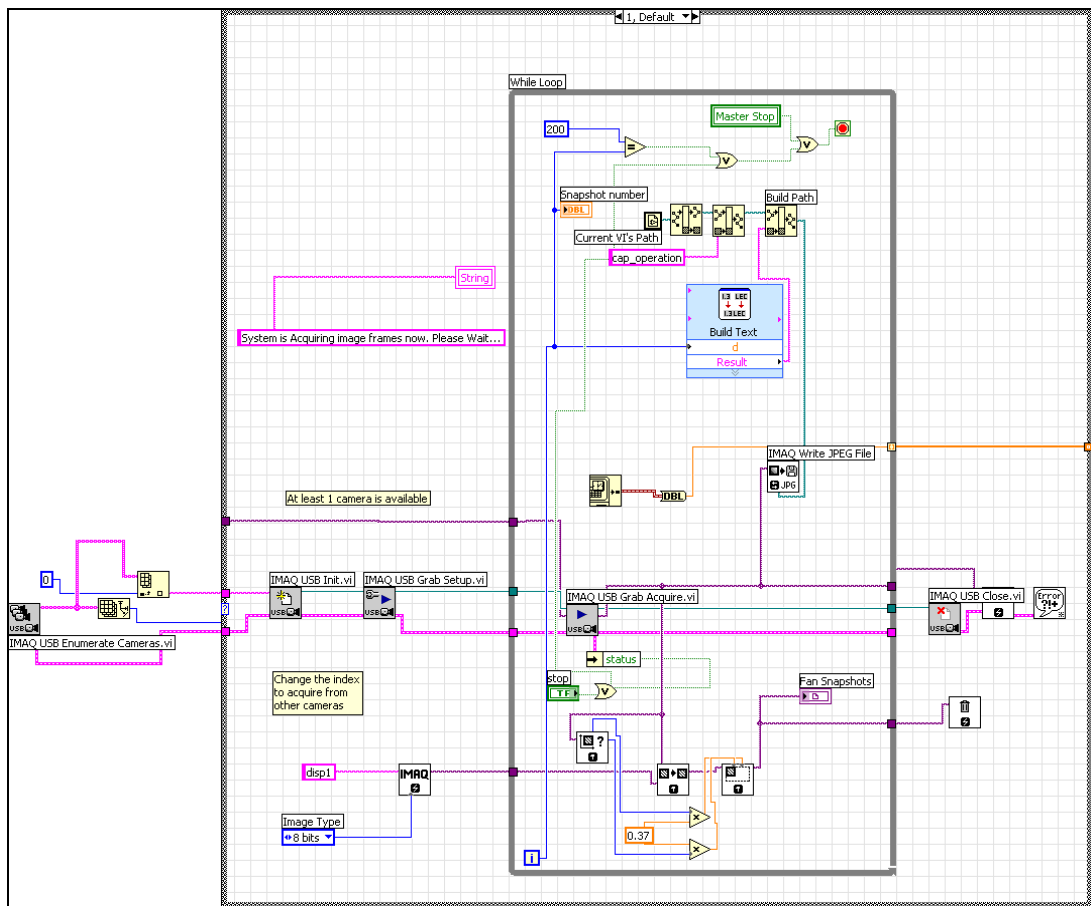


Figure 3.20 Image Acquisition LabVIEW Block Diagram

3.3.3.2 Step 2: Time Stamping

Actually, this step is included in the frame capturing stage. However, for the significance of this step, it is been left as a separate step of the algorithm. This step goes as:

As been mentioned in **Step 1**, a loop is computed for capturing of a preset number of frames. The loop itself saves the time stamps of every capturing.

At the end of capturing loop, the whole collection of time stamps of each frame is put into a 1D array to be used in another place in the program.

Figure 3.21 is showing the block diagram of this process.

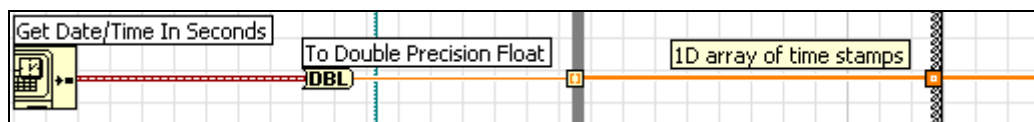


Figure 3.19 Time Stamping LabVIEW Block Diagram

3.3.3.3 Step 3: Frame Retrieving & Zooming

After finishing the phase of image acquisition, image are retrieved from the temporary place and zoomed to a desired area. This is all performed through:

1- Open the saved frames one by one in every iteration as shown in figure 3.21.

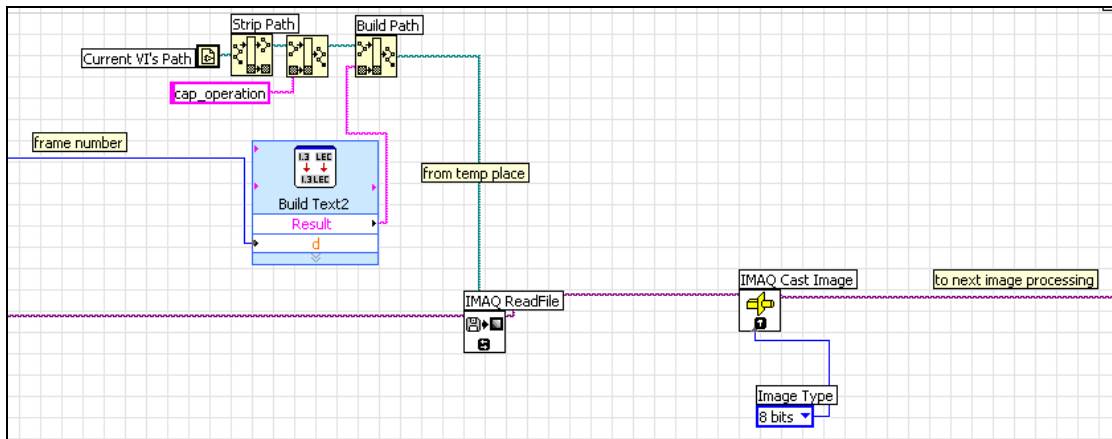


Figure 3.20 Retrieving Saved Image Frames LabVIEW Block Diagram

2- Figure 3.22 also shows the conversion of images into an 8bit grayscale through the block “IMAQ Cast Image”

3- Each coming image frame is zoomed to the specified coordinates as been set through the front panel (refer to 3.3.5.1) and enlarged to a certain factor, as shown in figure 3.23.

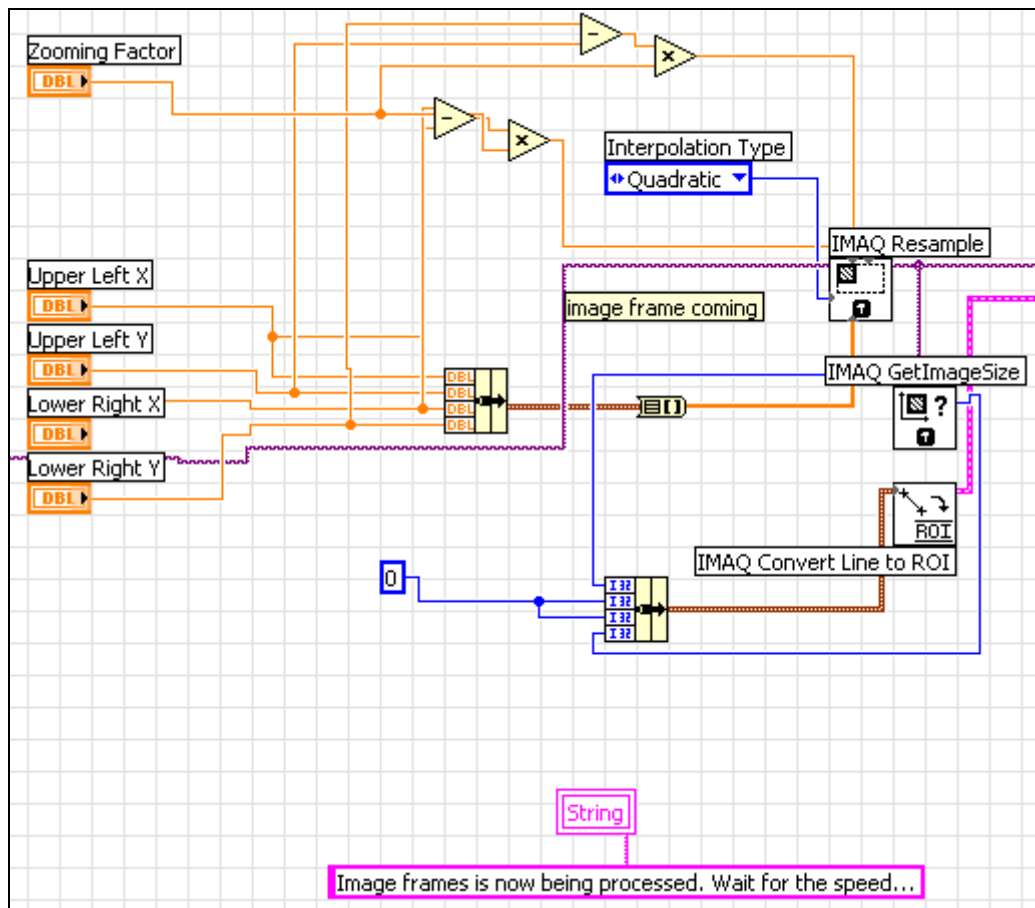


Figure 3.21 Zooming Area LabVIEW Block Diagram

3.3.3.4 Step 4: Edge Detection

In this step, the blocks from the group of ‘Machine Vision’ in vision toolkit of LabVIEW utilize their power. Every iteration, this step goes:

- 1- Receive the zoomed area from previous step
- 2- Construct a line profile (explained in 3.2)
- 3- Take edge detection parameters (shown in ‘User Interface’ and discussed in ‘Algorithm Tuning’ sections of the report).

4- From figure 3.21, “IMAQ Convert Line to ROI” block is to set the set of pixels to be investigated for edge detection. The standard line for the project is the line starting from the upper right corner of the image to the lower left corner (assuming optimal 1D line to track blade edges).

4- Extract edge(s) found in the designated line.

5- Draw the edges in image view in front panel for display

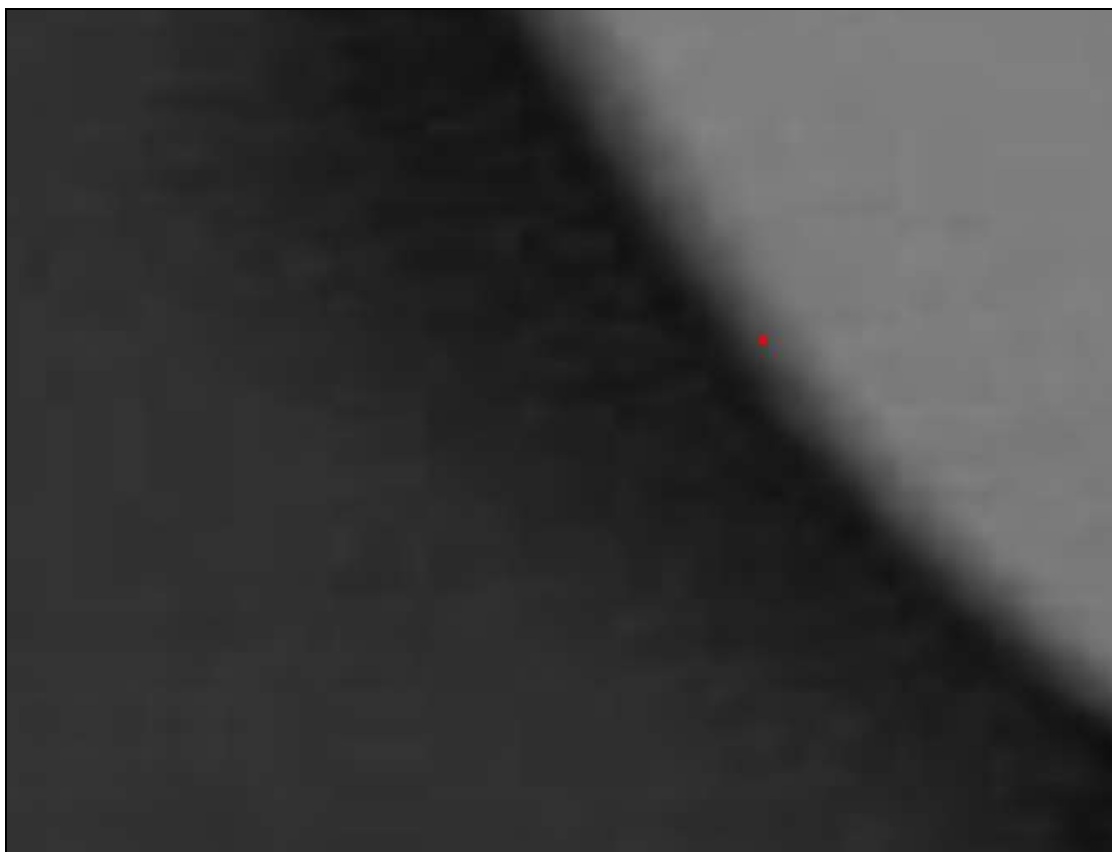


Figure 3.22 Example of Edge Detected

See figure 3.25 for this step in LabVIEW. This step concludes the image manipulation.

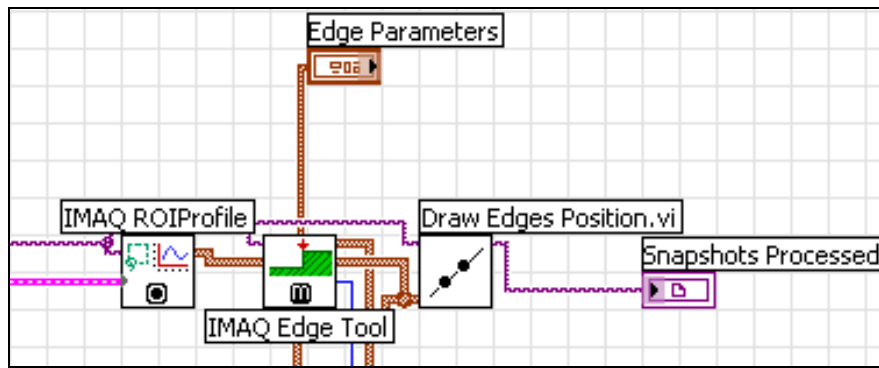


Figure 3.23 Edge Detection LabVIEW Block Diagram

3.3.3.5 Step 5: Extracting Edge Information

This step is an information process and extracting. After extracting all edges of a frame:

1- Edge(s) information is taken. The main information about the edges is: **edge polarity** and **number of edges found**.

2- In each iteration, only image frame with one edge is passed to next step (inside extracting edge information global step, i.e. Step 4). That is because of that the zoomed area is chosen carefully to capture a blade entering this area.

3- So, with one edge accepted, only an edge with a polarity of '-1' is accepted (i.e. light-to-dark edge or background-to-blade edge).

4- The previous step resulted in an initialization of a counter for number of blades passing the zoomed area.

5- Afterwards, the blade counter will not increment until a frame containing more than one edge (i.e. adjacent sides of consecutive blades) is detected.

6- The frame numbers of each accepted edge is saved.

7- The iterations in this step are stopped when counter reaches a preset number. In this algorithm, the number is 2 which means that two blades have passed.

The block diagram of this important step is shown in figure 3.26.

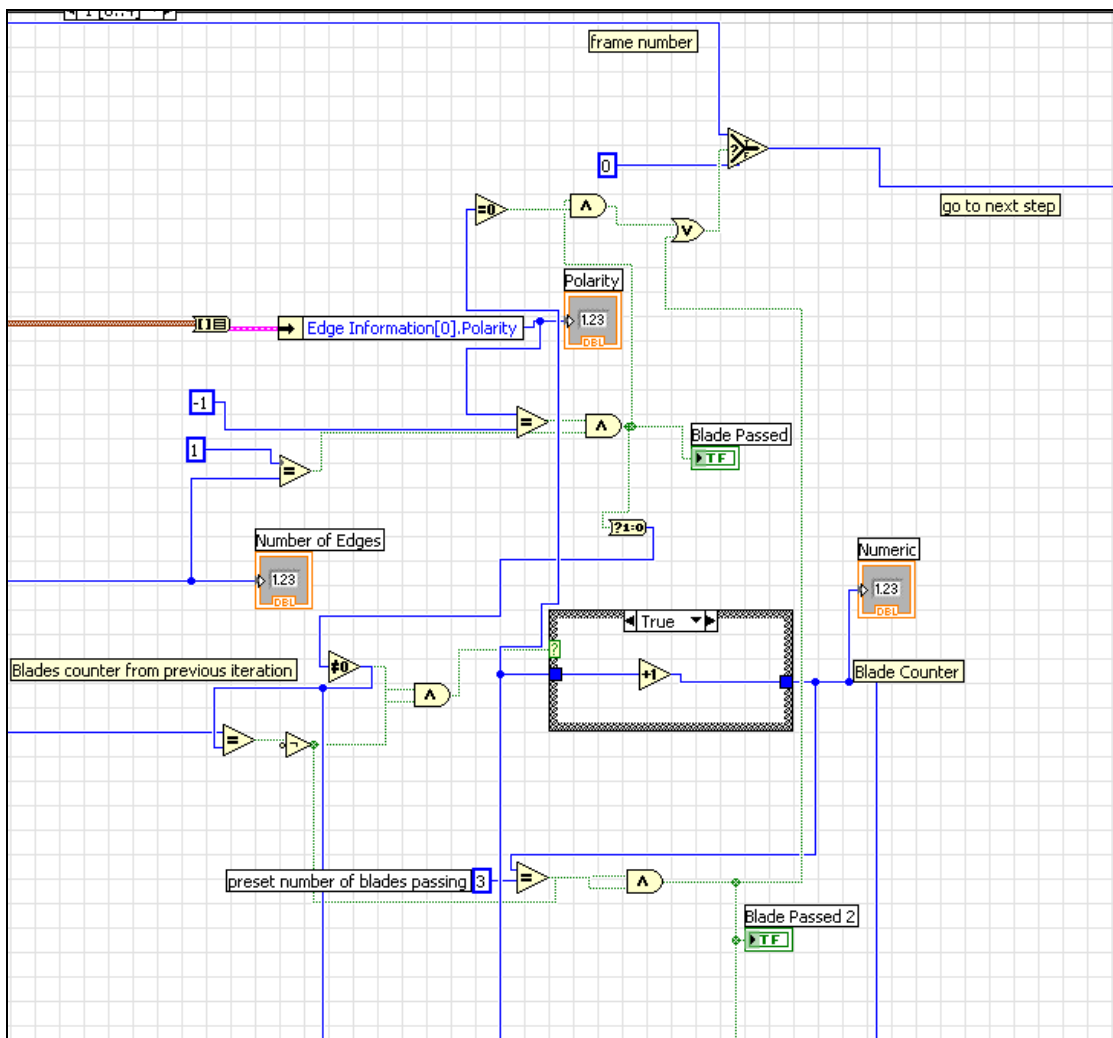


Figure 3.24 Extracting Edge Information LabVIEW Block Diagram

3.3.3.6 Step 6: Speed Calculation

The output of this would be the speed of the fan. This step computes through doing:

- 1- Retrieve the time stamps array from **Step 1**
- 2- Retrieve the accepted frame numbers array from **Step 5**
- 3- Take the difference in time between the time stamps of the selected frame numbers
- 4- Speed output in RPM is calculated.

$$RPM = \frac{60 \cdot n}{N \cdot \Delta t}$$

n = number of blades passed (parameter)

N = number of blades of the fan (input in front panel)

Δt = time difference between time stamps of the numbered frames

Figure 3.27 shows this part.

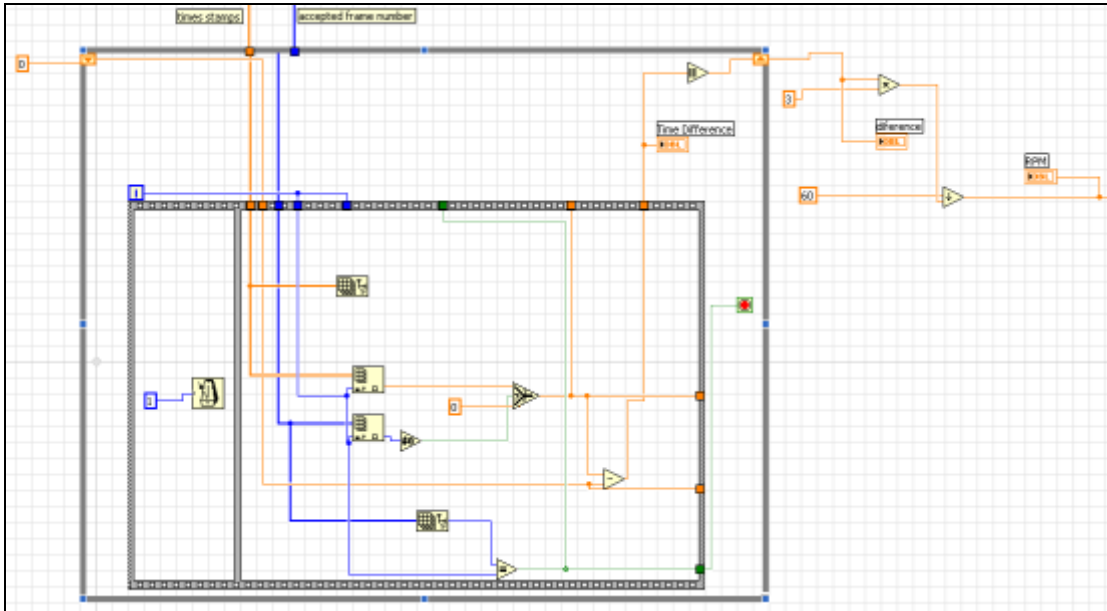


Figure 3.25 Speed Calculation LabVIEW Block Diagram

3.3.3.7 Step 7: Quality Decision

In this small step, only one computation is done. It is only verifies the value of RPM if it is in the range bounded by the Upper Limit and Lower Limit. Figure 3.28 shows the process.

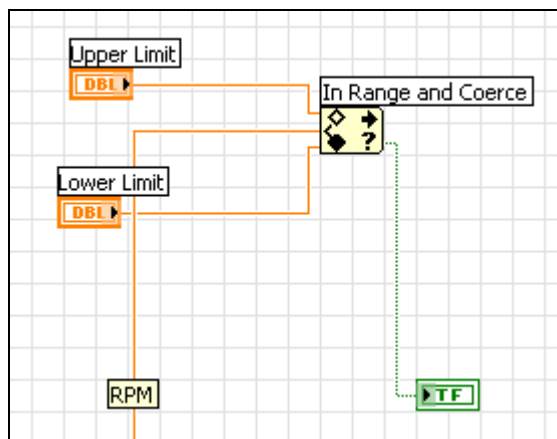


Figure 3.26 Quality Decision LabVIEW Block Diagram

3.3.4 Improved Algorithm

The first implemented algorithm suffers from an intrinsic drawback regardless of image noise and quality. This is due to the fact that a blade is detected in the zoomed region but can fall in a variety of positions or parts of the area leaving a relatively large area out of consideration.

To overcome this problem, some scrutinizing is required to either measure the angles between the blades or find a reference point –a variable reference point in our situation– in which to compare the position of the blades to give accurate measurements.

3.3.4.1 Algorithmic Options

3.3.4.1a Method 1: Angle Measurement

This method aims to measure angle differences from the knowledge of the edge detected coordinates. It turns out that this method is very difficult in our situation since the line where edges are detected identifies different areas of the blades as they rotate.

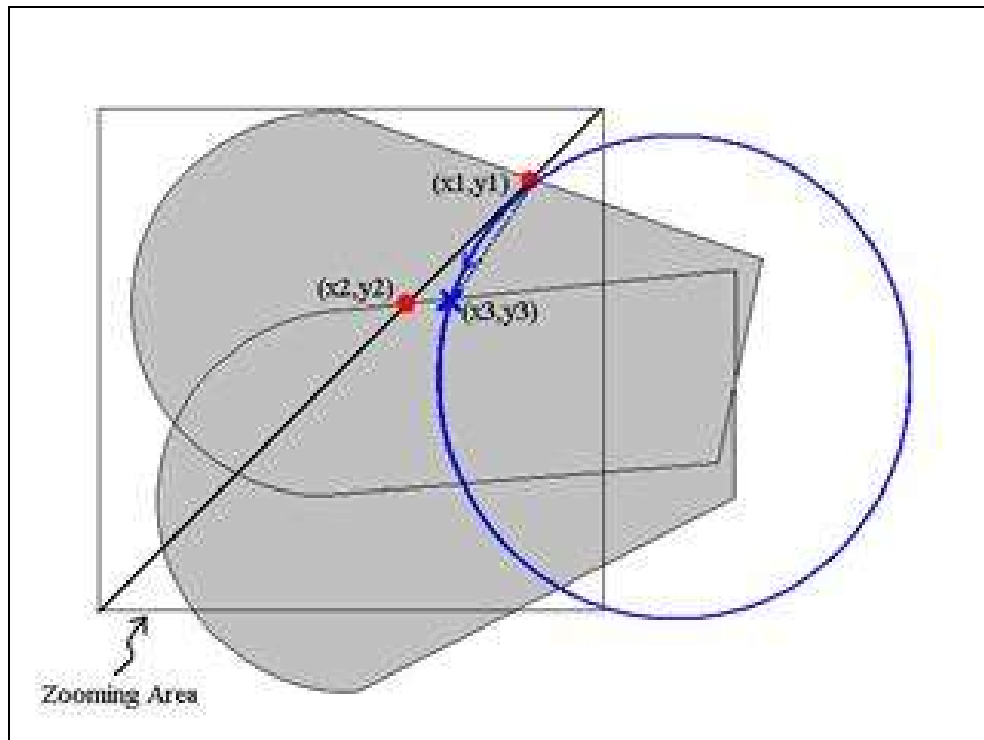


Figure 3.27 Angle Measurement Analysis

Referring to the figure above, we know the coordinates (x_1, y_1) and (x_2, y_2) . However, in order to compute angles accurately we must know the location of (x_3, y_3) since the angle $\theta_{(1)}$ it makes with (x_1, y_1) and the center of the fan is very different and even random compared to the angle $\theta_{(2)}$ between (x_1, y_1) , the center, and (x_2, y_2) .

Attempts were made to approximate the angle from the knowledge of (x_1, y_1) and (x_2, y_2) but our theoretical fears were confirmed as the computations showed that the measurement accuracy barely increased. The angles computed were also random as the difference angle $\phi = \theta_{(1)} - \theta_{(2)}$ also changed according to the blade positions and noise from the captured images.

Since LabVIEW does not have curved lines to detect edges on, the only solutions left are either to use several lines in grid or rotating format, which required an excessive amount of computation by that risking the reliability of the software, or pattern matching. Pattern matching was tried and proved to be very unreliable.

3.3.4.1b Method 2: Latching Coordinates

This method depends on latching the coordinates of the first detected edge and monitoring the subsequent edge coordinates while keeping track of the number of blades passed.

The edge detection technique is, as mentioned earlier, the same. However, instead of computing speed from two consecutive blades detected, a random number of blades will be detected each time the measurement will be conducted using this algorithm.

For practical reasons, the match in coordinates is not exact. An error margin (both in x and y axes) is specified by the user, and hence a comparison is done between the latched variable and every blade detected (not edge) to check whether the difference is within a tolerable value. If so, the number of blades passed during this monitoring process is divided on the time difference between the detection of the first blade and the blades with matching coordinates.

The only drawback in this method is that a compromise is made between accuracy and average number of images required to measure the speed. In demanding higher accuracy of the measurement, the user will have to tighten the error margin, which will make it more difficult to find a match and hence require more trials to be done by scanning more images on average.

3.3.4.2 Implemented Improved Algorithm

The selected new algorithm is **Method 2: Latching Coordinates**. As been described, this will lead to minor changes to the LabVIEW program.

For the improved algorithm, the steps: *Frame Capturing, Time Stamping, Retrieving Frames & Zooming, Edge Detection and Quality Decision* are all the same here. The changes apply to ‘Extracting Edge Information’ and ‘Speed Calculation’. The Changes are as follows:

3.3.4.2a Step 5: Extracting Edge Information

The step goes, after obtaining **Step4: Edge Detection**, as follows:

- 1- Edge(s) information is taken. The main information about the edges is: edge polarity and number of edges found.
- 2- In each iteration, only image frame with one edge is passed to next step (inside extracting edge information global step, i.e. Step 4). That is because

of that the zoomed area is chosen carefully to capture a blade entering this area.

3- So, with one edge accepted, only an edge with a polarity of '-1' is accepted (i.e. light-to-dark edge or background-to-blade edge).

4- From the first accepted edge, the location of the pixel is **latched** and saved.

5- The previous step resulted in an initialization of a counter for number of blades passing the zoomed area.

6- Afterwards, the blade counter will not increment until a frame containing more than one edge (i.e. adjacent sides of consecutive blades) is detected.

7- In each iteration, in each accepted edge, the x- and y-coordinates of the edge pixel are compared with the latched ones.

8- If the difference between the two coordinates is less than the mismatch tolerance, the counter stops and frame number saved. The counter value is saved also

The block diagram of this step is shown in figure 3.30.

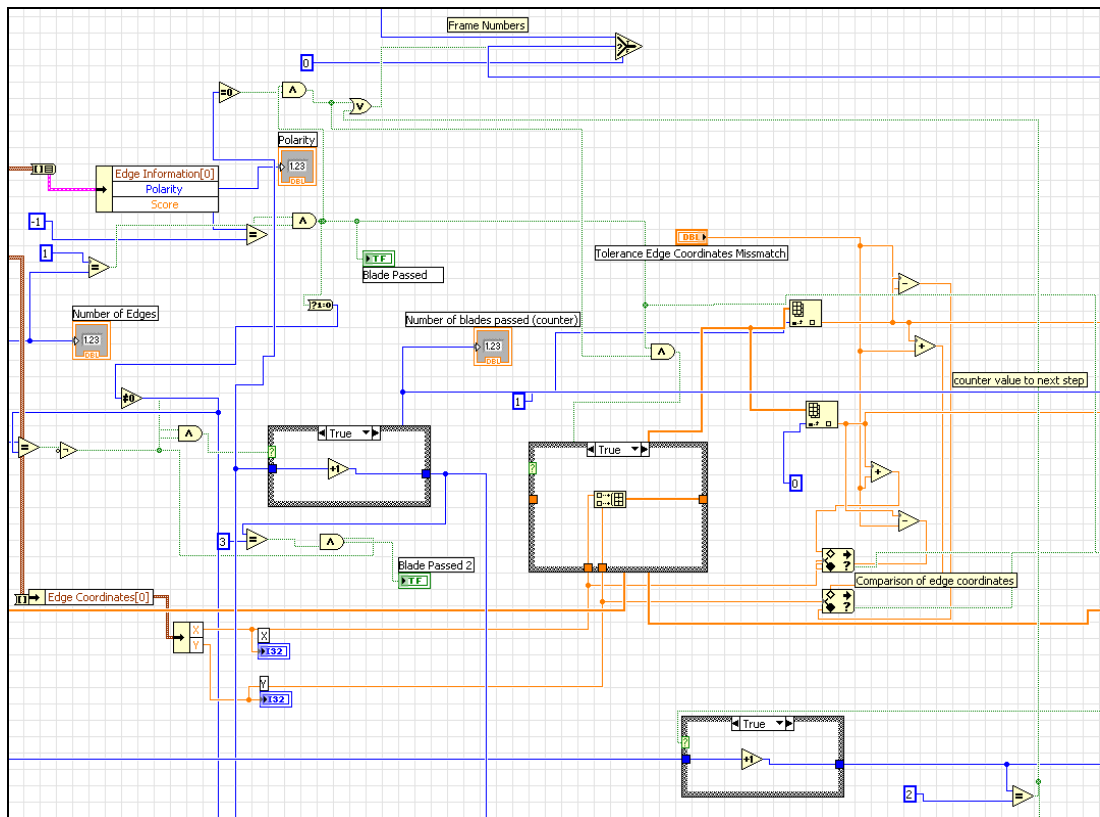


Figure 3.30 Improved Extracting Edge Information LabVIEW Block Diagram

3.3.4.2b Step 6: Speed Calculation

After getting the counter value and frame numbers from previous step, this one goes as follows:

- 1- Retrieve the time stamps array from **Step 1**
- 2- Retrieve the accepted frame numbers array from **Step 5**
- 3- Take the difference in time between the time stamps of the selected frame numbers
- 4- Speed output in RPM is calculated:

$$RPM = \frac{60 \cdot n}{N \cdot \Delta t}$$

As,

n = number of blades passed (variable output from previous step)

N = number of blades of the fan (input in front panel)

Δt = time difference between time stamps of the numbered frames

Figure 3.31 shows this part.

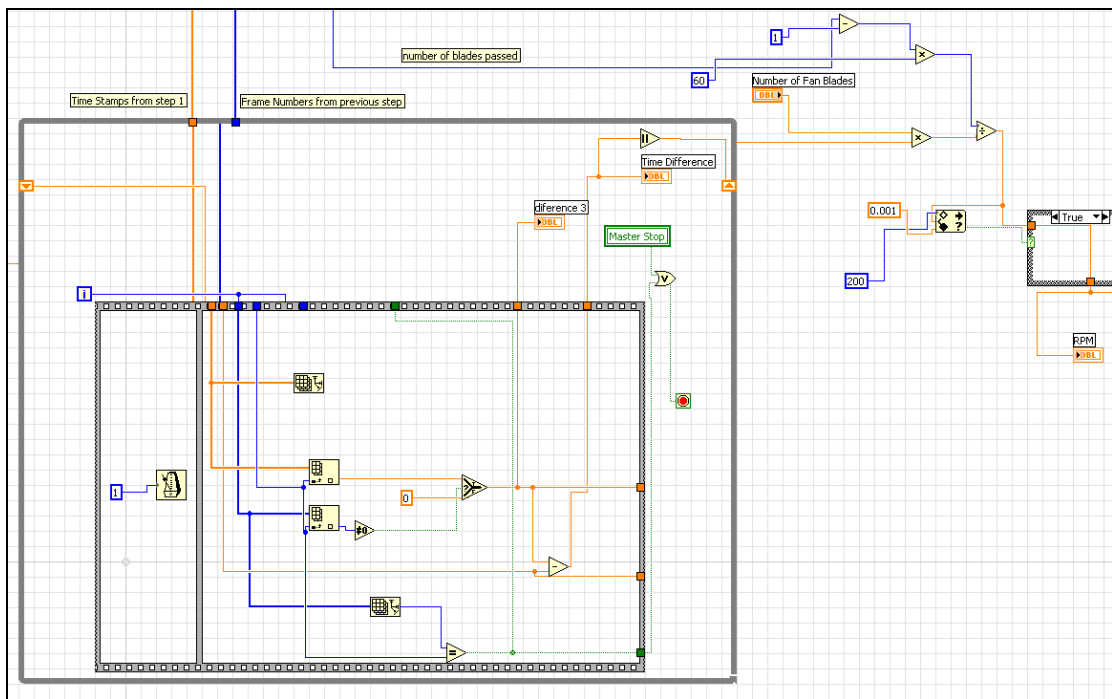


Figure 3.28 Improved Speed Calculation LabVIEW Block Diagram

3.3.5 Software Outputs

This section is going to show the end-product's outputs, or in other words outcomes. For this project, the outputs are the ones which are coming out of the software. The software program implemented has two main 'outputs'. These outputs can be divided into:

- User Interface
- Automatic Decision

3.3.5.1 User Interface

This section will show the User Interface of the software that is measuring the fan speed. The User Interface function is not only to display the result of the speed. The complete description of the User Interface will be mentioned in the next paragraph.

Both figure 3.32 and 3.33 are showing the front panel. The front panel of the user interface for the operator contains two tab page titled as "Operation" and "Advanced" respectively as show in the top of both figures.

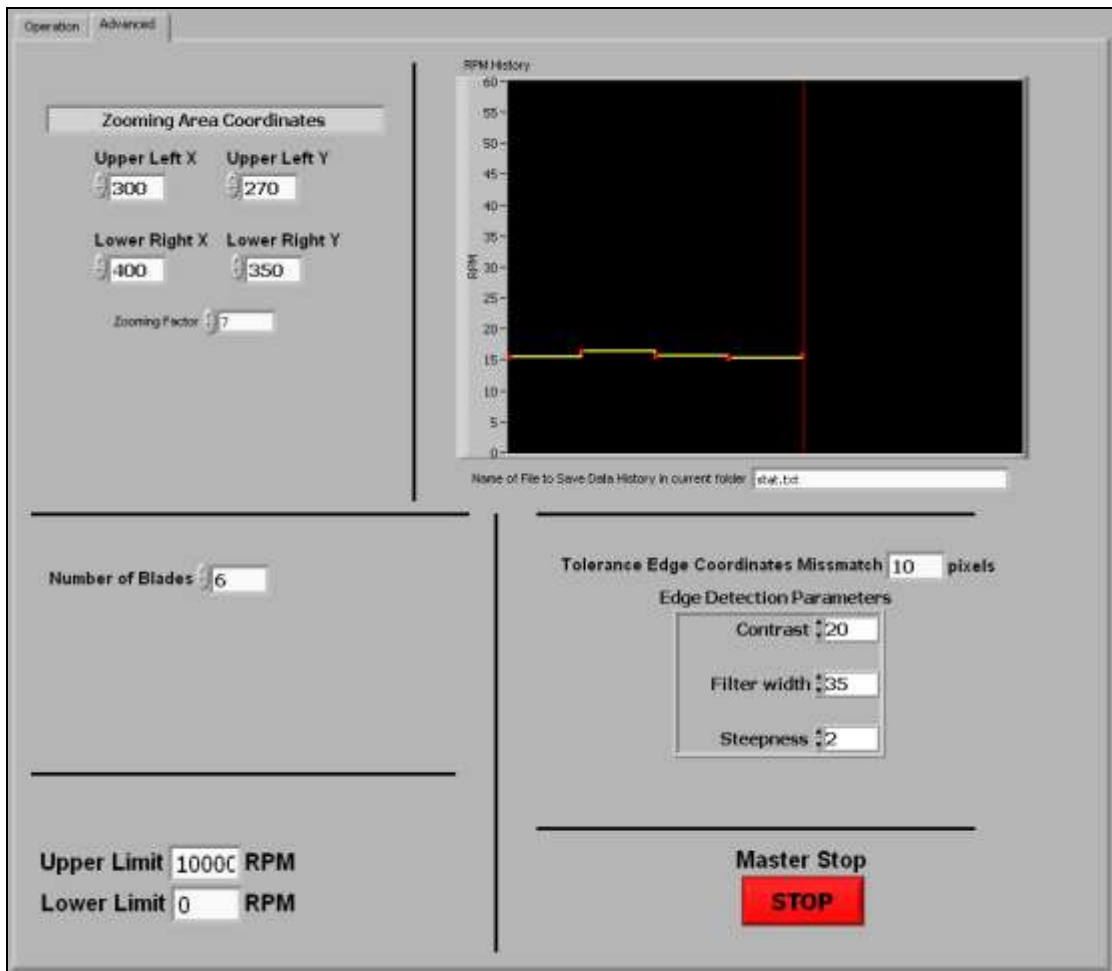


Figure 3.30 'Advanced' page of User Interface

In figure 3.32, the “**Operation**” page of the front panel is shown. As displayed, it contains:

- *Left upper corner window* shows the live snapshots of the fan being acquired before the processing stage. The number reflects the n th frame being captured. A “Stop” button is also there for stopping the acquisition at any time and let processing continues to the next phase.
- *To the right of the live snapshots window*, the speed of the fan is shown which is updated every iteration in the unit of RPM (Revolution per Minute). The number of the frames being processed is also displayed.

- *Above the RPM value, the title toggles between “System is Acquiring image frames now. Please Wait...” and “Image frames is now being processed. Wait for the speed...” alternatively.*
- *The circle to the right is indicating the “pass” of “fail” status of the fan. It is showing “Pass” in the figure 3.31.*
- *Below the pass-or-fail indicator, data history of all fan being processed or previously processed is displayed also. The table is associated with Mean and Variance fan production batches.*

In figure 3.33, the “**Advanced**” page of the front panel is shown. This page is used for changing a lot of parameters regarding the program. The page contains:

- *As titled “Zooming Area Coordinates”, this section is responsible for the coordinates of the zooming area that chosen to be processed. It consists of the x- and y-coordinates of both the upper left corner and lower right corner as indicated. This section also contains the adjustable zooming factor of the area to be processed.*
- *To the right, a graph of 5 previous history values are displayed and updated. This section also contains a field for writing the name of file to store and retrieve the historical data.*
- *Below the zooming coordinates section, a section of the fan parameters. Only the “number of blades” is required.*
- *To the right, parameters related to the edge detection processing are shown. A numerical field is left for the tolerance allowed for edge*

pixel position mismatch. The other fields are concerned with specific edge detection algorithm parameters.

- The “*Upper Limit*” and “*Lower Limit*” values are to be set by the facility that uses the software to test speed of the fan. These values are deciding the “pass-or-fail” status.
- at the very lower right corner, a “*Master Stop*” button is used to shut off the whole program and stop all operating tasks of the program.

As been mentioned earlier, another output of the program is the history data file that is saved in the PC’s hard drive. As shown here, the name of the file is “**stat.txt**”.

3.3.5.2 Automatic Decision

Another ‘output’ of the system is the automatic decision of “Pass” or “Fail” for the tested fan. Other than the screen indicator of passing or failing, the indicator can be used as a trigger for further actions that to be taken in both situations of “Pass” or “Fail”. Different kinds of actions resulted will be mentioned in the “Recommendations” part of the report.

3.4 Physical System Design

After developing the software, suggestions are made here for the installation of an actual system in the plant. This involves tuning the measurement and idle time periods and the appropriate positioning of the camera.

3.4.1 Time Synchronization

Given the inter-arrival time between the Air Conditioners to the position facing the camera, the software can be assigned a specific amount of delay between each measurement process.

This, however, is no doubt a crude method of synchronization of the measurement system with the physical process. It assumes the inter-arrival time is deterministic and fixed although this not absolutely the case. Nevertheless, the value has low variability which justifies fixing the parameter, although the system remains blind to changes since no feedback is present.

A simple method to include feedback to the synchronization process relies on positioning trip switches in appropriate areas in the vicinity of the conveyer belt; the output of the switches can easily be fed to the software to trigger the measurement. Refer to 5.3 for recommendation on using more advanced techniques using image processing.

3.4.2 Camera Mounting

The position of the camera depends heavily on the equipment positions surrounding the air conditioners. It should be selected wisely as to avoid interference with obstacles such as workers' hands, tools, and instruments.

From the information obtained on the plant involved and the appropriate angles of which the software can handle with low probability of error, it seems best to mount the camera horizontally, a few centimeters above the level of the conveyer belt.

4. RESULTS AND ANALYSIS

4.1 Algorithm Tuning

This part will contain some discussion about the effect of parameters on system performance. Most of the discussion is about the parameters of the image processing operations of the software; namely, Edge Parameters and Zooming Area.

4.1.1 Parameters

As explained, this section is about the effect of software parameters on the over all system performance. Software parameters are concerned mostly to the image processing techniques, specifically the edge detection parameters. Edge Detection parameters are:

1- **Contrast:** as been mentioned earlier in both “Software Development” and “Image Processing” sections of the report, Edges in image depend on the difference in gray level. Contrast can be equivalent to this difference. So, the value of *Contrast* to be set is dependable upon the color of fan and the

background. Generally, Contrast value is adjusted to fit the gray level values differences of edges taking into consideration the ambient effects.

2- **Filter width:** as been explained in “Image Processing” section, edge detection can be implemented on 1D or 2D values. In the project method, it is only 1D. So, a line of pixels is investigated for edges. In this line, “filter width” means the other pixels width around each pixel point at a time which is being considered as area for searching for a difference in gray level values. It has been observed that filter width values are highly depending on the blur around the image edges. So, values are adjusted to let a space for searching for blurred edges.

3- **Steepness:** one of edge parameters is the difference of two gray levels, but the slope of this occurring difference also plays a role. Steepness can be analogous to this slope. Go to figure 3.14 to refresh the concept of edges in theory.

4.1.2 Zooming Area

The Zooming Area is a part of the image that is zoomed-in for processing. So, the coordinates of this area is to be selected appropriately to achieve the expected outcomes. Surely, the area is expected to be passed continuously by the fan blades to enable the edge detection. It has been observed that as more the area is near the center of the fan as the opportunity of high performance of the system is expected. This is due to the reason of ‘slow’

speed of picture change (i.e. blade motion) rather than ‘fast’ one near the outer shell of the blades.

4.2 Experimental and Statistical Analysis

The results of image processing are not exact by the nature of its operands – images–, especially regarding problems related to *detection* and *estimation*. Even though the equations involved are normally deterministic, the object of analysis is inseparable from noise and sources of randomness.

For example, in two consecutive images captured of the same objects, the edge detected separating two objects apart does not generally match in the two images (the coordinates are correlated but not the same), even though the filter parameters are fixed.

By the same token, even though the algorithm designed in this project is deterministic, there are two main sources of randomness: the first in the form of noise in the captured images, and the second from the variable frame rate of the camera. Hence exact measurements were not sought from the beginning as image processing was considered. This goes in harmony with the requirements of the real system, namely, ensuring that speed falls in a certain –relatively large- range.

And as the situation is as such, statistical analysis is indispensable in gauging the performance of the measurement system.

4.2.1 Collecting Samples

A function labeled 'write to spreadsheet' was used to collect continuous readings of the measured rpm and store them in a text file. After the completion of both programs 1 and 2, a standard data collection experiment was performed. The organization and type of data collected is rationalized in the objectives of the statistical analysis mentioned in the subsequent section.

Given that the speed range suitable for the prototypical hardware was roughly between 0 and 60 rpm, the fan speed was fixed at five consecutive values 10, 20, up to 50 rpm (with the optical tachometer as reference) and at each of these mentioned speeds, the two programs were run repeatedly and their outputs were recorded. Hence assuming 100 outputs were measured for each program for each fixed speed, a total of $2 \times 5 \times 100 = 1000$ measurement outputs were collected.

For example, these are the first 22 readings of the first program measuring the speed of the fan at reference speed of 10 rpm:

Table 4-1 few Sample Data

10.157	11.031
10.411	9.625
9.77	10.241
10.32	10.406
10.081	9.625
9.926	10.667
9.921	9.926
11.534	10.667
9.921	11.848
9.847	11.13
9.847	9.921

For the rest of data collected, go to Appendix C.

4.2.2 Distributions and Statistical Analysis

The objectives of the statistical tests are the following:

A. Validating that the average of the measurement outcome of the two programs confirms to the reference measurement. This means that although the measurement sequence will randomly fall slightly above and below the reference value, the sequence average will converge to the reference as a mean. This *ensures there is no consistent bias in the measurement.*

B. Confirming that the variance in the measurement is within a tight acceptable interval, by that ensuring confidence in the output measurement and the decrease in probability of large errors.

C. Comparing the results of the first and second programs, in hope of finding improvement in the second by showing reduction in variance.

D. Observing the effect of speed on the accuracy of the measurements (mean and variance), by this answering the question of to what extent measurement reliability is dependent on different points in the measured range.

In effect, data collection was done in reference to these objectives in mind. Although the team wished to take more samples, this was constrained by the relatively long time period it took the webcam to capture frames. (The webcam took an average of about 20 frames per second, and each of the 1000 output measurements took an average of 100 image captures to compute a speed estimate, hence roughly 100,000 images were captured for this test taking 5000 seconds for image capture alone, not considering operating and computation times). Nevertheless, the number collected of about 1000 seems to be sufficient for non-rigorous analysis.

Data points pertaining to the specific reference speeds and program number were analyzed individually from their text files automatically generated by LabVIEW. These are ten text files containing all combinations of programs and speeds (refer to previous section on data collection).

A professional statistical software package was used, namely, ***Rockwell's Arena Input Analyzer***. This Input Analyzer can read a text file and perform a variety of statistical tests –upon user's prompts– including plotting the data in a Histogram and performing best fit computations.

The ten files were fed to this software, each generating a histogram. The shape of the data was found to be –as was logically expected– roughly symmetric about a mean and similar to the bell shape of the normal distribution. Some of the data files, however, fit the normal distribution better than the others.

A point must be clarified here to alarm the reader from misleading results of data fitting software. The team has previously studied a course on simulation of discrete event system and has gained experience in data fitting. The authors of the textbook studied stressed this fact:

The physical and logical nature of the process of which samples are drawn from must be considered when selecting a probability distribution. Some complicated distributions are very flexible with their large number of parameters, and might hence surpass logically sounder distributions in data fitting software, especially when the number of samples is not very large.

For example, arrival rates are logically and ordinarily modeled with exponential distributions. However, given data corresponding to arrival times and requested to find the best distribution, the data fitting software usually doesn't select an exponential distribution, but rather a more complex distributions like Beta, Gamma, or Weibel, of which the exponential is a special case of.

What's more, with a relatively low number of samples, the software selection of best fit is extremely sensitive. It was observed in some cases that changing one data point by a decimal value changed the optimal distribution computed by the software from Beta to Weibel.

This hopefully validates the decision of comparing the data to a normal distribution, which was initially assumed in the measurement system since no bias existed in any phase of the measurement.

The reader is referred to the appendix for the complete graphs and results of the input analyzer. The following table summarizes the most important information, namely, the mean and variance of the normal fit of the data:

(μ, σ^2)

Table 4-2 Mean and Variance values

	10 Rpm	20 Rpm	30 Rpm	40 Rpm	50,54 Rpm
Method 1	(10.1, 0.647)	(20.8, 1.52)	(30.2, 2.31)	(40.5, 3.69)	(49.9, 4.25)
Method 2	(10.2, 0.714)	(21, 2.29)	(29.8, 2.27)	(40.9, 2.5)	(53.2, 3.08)

Outliers which were not the result of the logical structure of the algorithm but rather due to low image capture quality could be eliminated by taking a *vote of the average values*. For example, assume the figure 4.1 below represents a relative experiment where six measurements are performed for the same reference speed. If the probability of *not* being an outlier is over 50%, then as more values are taken and averaged, the easier it is to identify a certain range and eliminate the erroneous outlying values

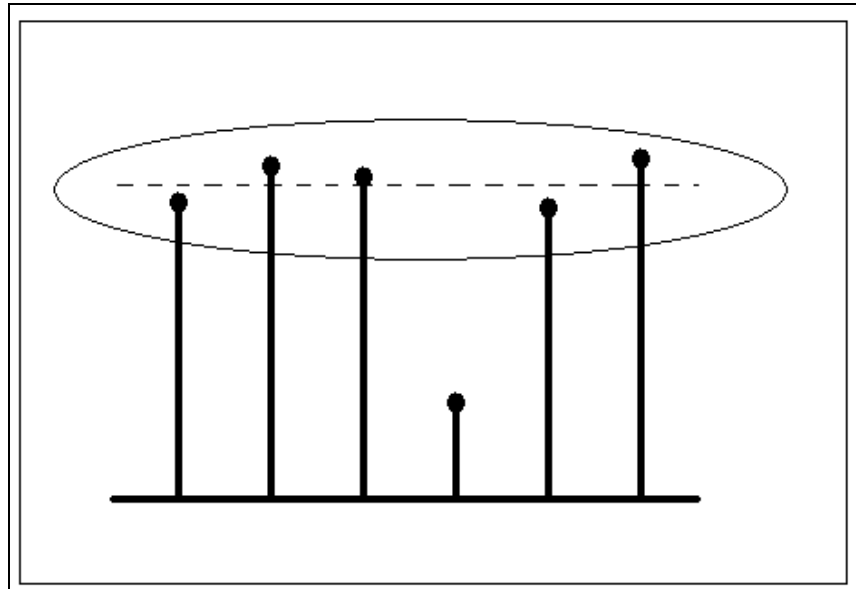


Figure 4.1 Illustration about rejecting Outliers

4.2.3 Output Statistical Analysis

The output analysis is done in light of the statistical analysis objectives stated in the previous section and the data collected with its summary in the table 4-2 above.

The average of all the ten experiments shows the absence of bias in the measurement. All of the averages are almost exactly centered at the reference speed.

The variance is also relatively small, giving an error in the ranges of 6 to 9% in the first program and 5 to 10% in the second. This however does not mean the second program has higher variability at some speeds than the first

program. The problem is with the noise in the images and miscounting edges. Although considering the logical functioning of the algorithms and assuming ideal edge detection, the second program has very high and controllable accuracy (by controlling the tolerance of coordinate mismatch) and would, under ideal cases, provided almost exact measurement.

Nevertheless, the table captures a significant point. Notice the variability increases proportionally in both programs as the speed reference increases. This is expected, as the quality of the captured image degrades as the speed increases, which is a problem caused by the low frame rate of the webcam.

The table also shows that the variability is significantly lower in the second program as the reference speed increases, an encouraging fact indeed, since the team feared this theoretical necessity would not show in the sample due to the image noise and miscalculation of number of blades.

4.3 Meeting Objectives

This brief section is an assessment of the project output based on the statistical analysis conducted. The assessment of success is relative to the objectives mentioned in 2.4.

The measurement system is reliable in the sense that one normally gets a very close value to the reference measurement and can insure correct results

at least by taking an average of several values. Even outliers caused by low image quality can be eliminated as shown previously. Hence the term ‘measurement system’ can safely be attributed to the project.

A full –although prototypical– software package was also designed for the end user. Along with the equipment required, this software provides a fully automatic measurement system, performing tasks such as reading values, testing for failure and success, and storing data.

The objective of designing an innovative system utilizing techniques of image processing was also clearly met. It also proves the wide applicability of image processing and opens new doorways for further applications.

5. CONCLUSION

One of the most significant conclusions related to the algorithms and project cost is this. Using edge detection to measure rotational speed requires a frame rate proportional to the number of blades of the fan, while mark extraction is independent of the number of blades. This should be considered when purchasing the industrial camera as explained in 3.1.1.

The following sections of the conclusion are related to the overall characteristics of the project:

5.1 Project Innovation

One of the main characteristics of this project as a whole was its affinity towards innovative ideas, including both new objectives (ends) and new design techniques (means). This property justifies a lot of the apparently unreasonable efforts and costs required to develop the system.

It also justifies the relatively long period of time it took the team to complete the project: there were simply no previous attempts nor reports to indicate whether ideas and attempted solutions were going in the right direction or not in many of the vital decisions of the project.

Although image processing techniques in general are commonly known and used in applications, and although rotational speed measurement systems are also abundant in industry, applying the methods of the former to fulfill the requirements of the latter was almost never previously attempted. A significant aspect of the project was its provision of this link.

In part 2.3, several initial justifications of vision processing were provided. Now, in the light of the results obtained from the actual project, it remains up to the reader to assess whether these justifications were satisfying or not.

5.2 Applied Recent Technology

This Project has exploited many aspects of relatively recent technology. On one hand, the project requires –in its industrial implementation– a high performance camera with technical specifications. This provided the team an opportunity to learn up-to-date technologies related to industrial cameras. What's more, the project is strongly dependent in its being actually implemented on rapid development of camera technology, which will provide higher frame rates and less noise at lower costs, by that reducing the price of this measurement system and increasing its performance.

On the other hand, the utilization of image processing techniques in 'semi' real-time applications is also relatively recent. Image processing algorithms

are computationally expensive and only recently have PCs gained the speed –at reasonable costs– to execute such algorithms in real time.

5.3 Recommendations

The team encountered suggestions of adding a control strategy based on the information provided by the measurement system they have designed. However, the measurement system in this project is not part of an automatic control system, but rather of a quality control and monitoring system. Keeping this in mind, the recommendations offered will be oriented towards this industrial application, even though the team is aware of general extensions like fan speed control following fan speed measurement.

- **The first recommendation** is related to increasing the robustness and general applicability of the measurement system. The software developed in this project relies on time synchronization between the measurement process and the arriving fan. Even though the sequencing can be improved by a very basic idea composed of placing trip switches on the conveyer belt and connecting them to the software as measurement flags, a more sophisticated method based on the image processing measurement system itself can be further developed.

For example, an algorithm can be designed to recognize a certain aspect of an arriving AC to initialize the image capturing of the rotating fan. This would be useful in reducing the number of images captured by the camera

and avoiding misleading images that might effect the decision of the speed measurement algorithm. In effect, this would ultimately lead to lower energy consumption of the camera in operation, less computation required on behalf of the computer, and higher measurement reliability.

- **The second recommendation** is related to an idealization assumed in the process, namely, that the angle between the camera and the arriving fans is always fixed. This is only an approximation, although a good one. For increased robustness towards angle uncertainty, the team recommends a sophisticated algorithm that can deal with this angle as a random variable, and perform an inverse mapping of an image bearing inclined blades to an ideal image with zero inclination.

A step further might be an intelligent algorithm capable of distinguishing between varying light intensities and actual blades. This might be necessary in an active environment where several objects are in motion near the fan causing shades or reflections on the blades.

The second recommendation presents a different direction for a future continuing project. Its main emphasis is not on increasing the power of the current algorithms for higher reliability and robustness of rotational speed measurement, but rather on extending the range of functions of the system.

More specifically, since this system was designed to operate in a quality control laboratory, it would be very beneficial if the software could take

benefit of the more common application of image processing in this setting, namely, in visual inspection. Certain characteristics of the ACs could be assessed automatically via image processing, like searching for dimensional irregularities, damaged parts, missing parts, and also counting and statistical quality analysis.

This could also be extended to higher levels of autonomy if communication networks were involved. Number of ACs, their characteristics and defects can all be stored in databases and communicated with management or engineering divisions.

5.4 Report Conclusion

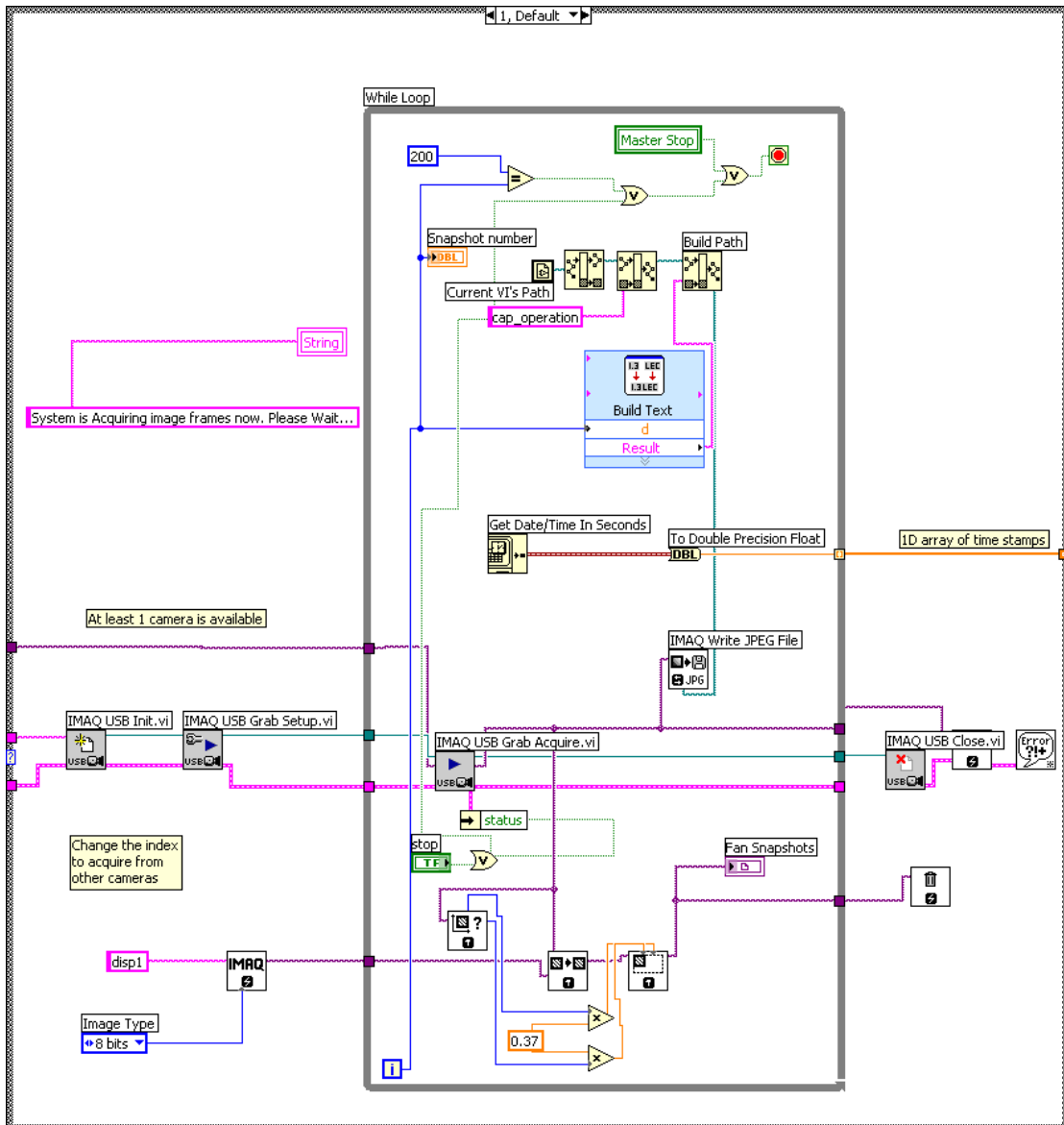
In this report, the team covered an in depth engineering design process, namely, an automatic rotational speed measurement system by means of image processing, for an industrial problem related to quality control. All phases of the design project were described in considerable length.

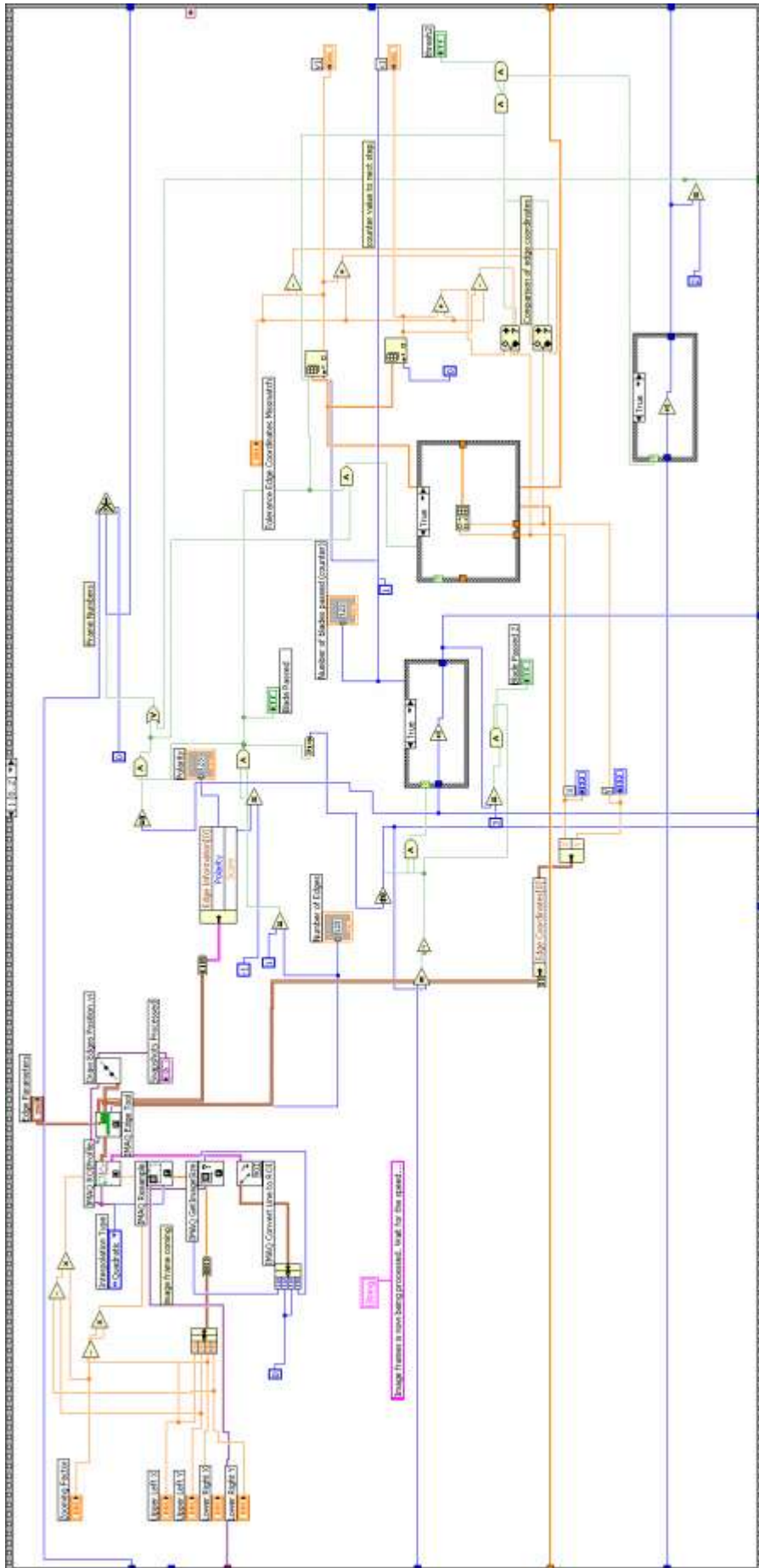
The team offered suggestions for further elaborating projects that can pick-up from where they left, hoping that a fully automatic and robust measurement system can eventually evolve.

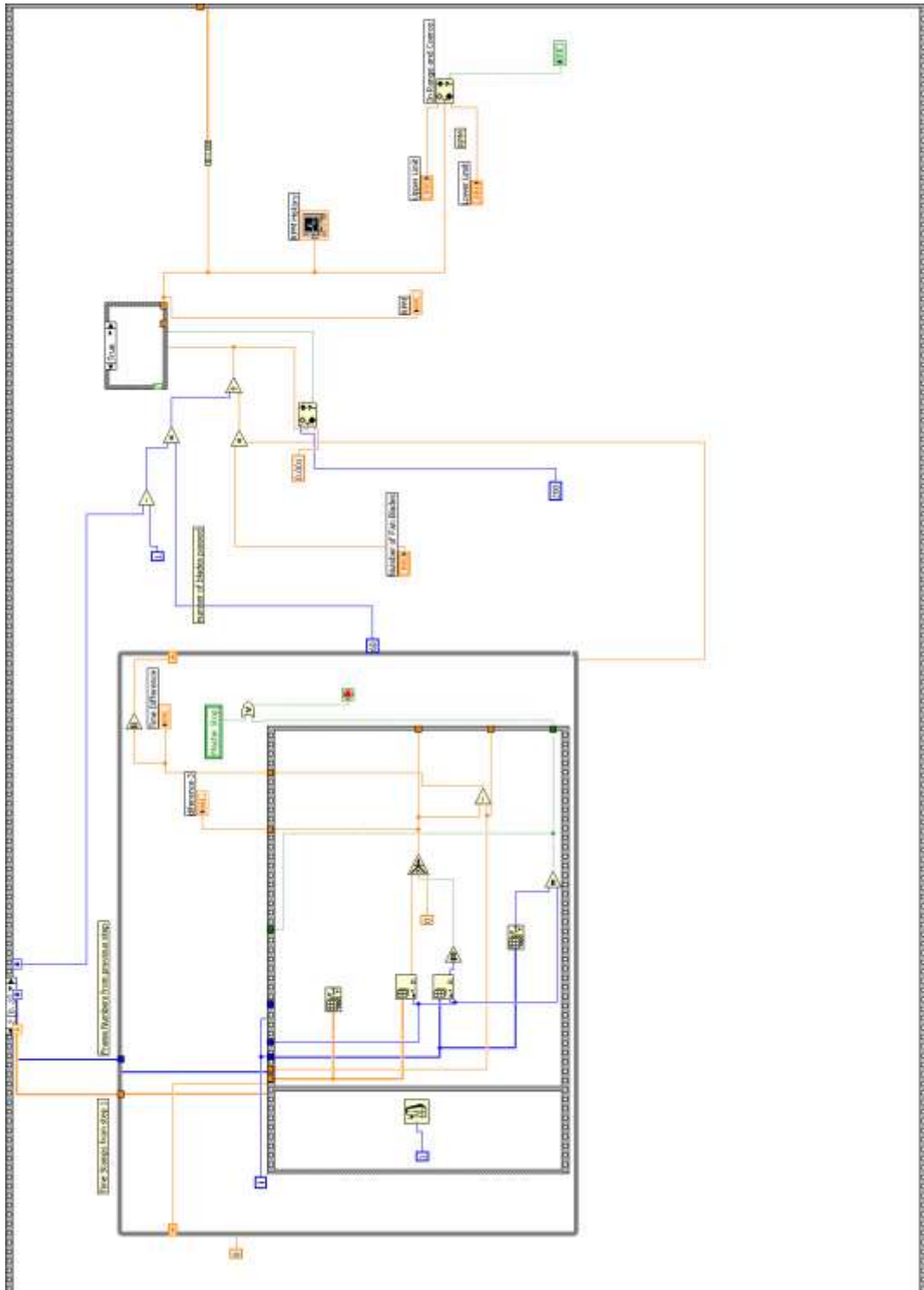
REFERENCES

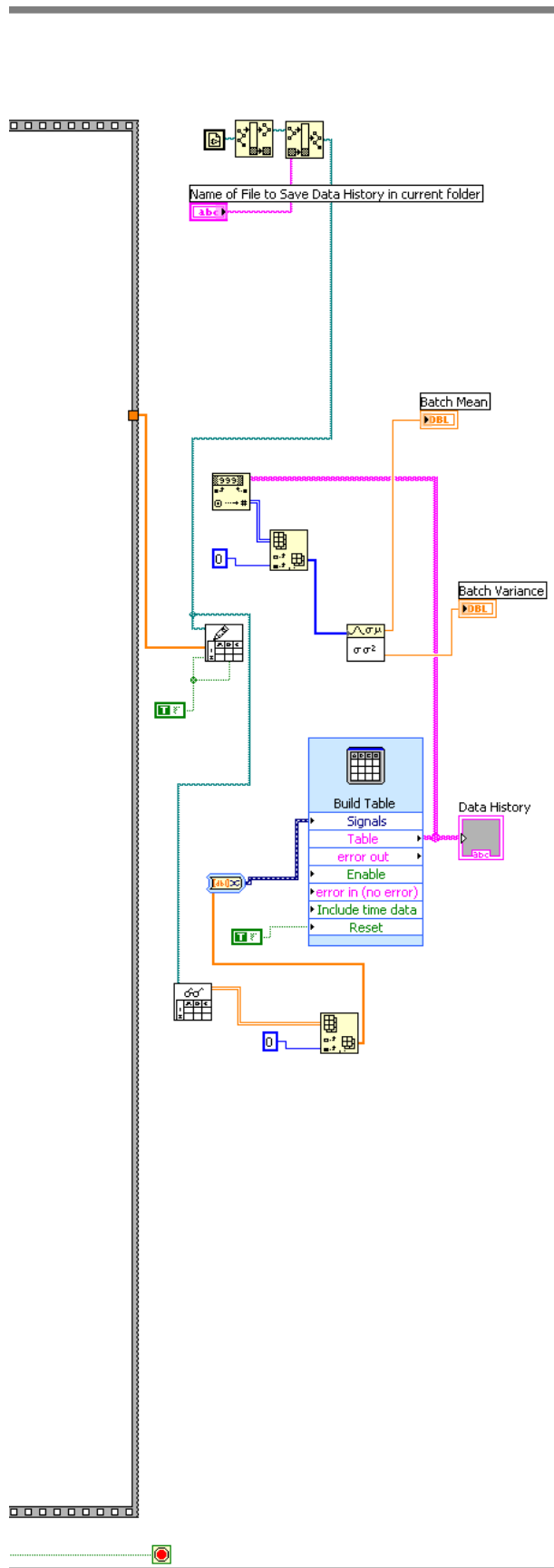
- 1) Douxchamp, Damien. The IEEE1394/Firewire Digital Camera List: scientific, industrial and other machine vision cameras.
<<http://damien.douxchamps.net/ieee1394/cameras/index.php>>.
- 2) Eastman Kodak Company. CHARGE-COUPLED DEVICE (CCD) IMAGE SENSORS. Primer. Rocheste: Eastman Kodak Company, 2001.
- 3) Encyclopaedia Britannica. tachometer -- Encyclopaedia Britannica.
<<http://www.britannica.com/eb/article-9070863/tachometer>>.
- 4) Fellers, Thomas J. and Michael W. Davidson. Molecular Expressions Microscopy Primer: Digital Imaging in Optical Microscopy - Concepts in Digital Imaging - Digital Camera Readout and Frame Rates. 2005.
<<http://micro.magnet.fsu.edu/primer/digitalimaging/concepts/readoutandfram erates.html>>.
- 5) Gonzalez, Rafael C. and Richard E. Woods. Digital Image Processing. New Jersey: Prentice-Hall Inc., 2001.
- 6) Litwiller, Dave. "CCD vs. CMOS: Facts and Fiction." PHOTONICS SPECTRA January 2001: 4.
- 7) Purcell, Andy. An Introduction to the Instrument and Industrial Control Protocol.
- 8) Universal Serial Bus - Wikipedia, the free encyclopedia.
<<http://en.wikipedia.org/wiki/USB>>.
- 9) Yoshihiro Watanabe, Takashi Komuro, Shingo Kagami and Masatoshi Ishikawa. "Real-time Visual Measurements Using High-speed Vision." (n.d.): 9.

APPENDIX A: LABVIEW FULL PROGRAM





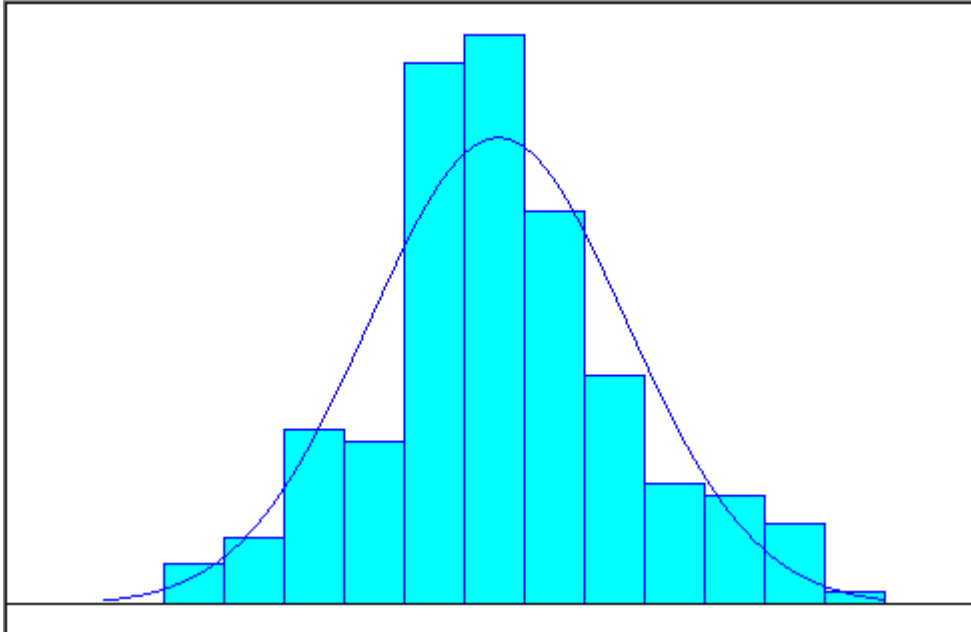




APPENDIX B: FULL STATISTICAL ANALYSIS OUTPUT

Program 1:

10 Rpm



Distribution Summary

Distribution: Normal
Expression: **NORM(10.1, 0.647)**
Square Error: 0.009521

Chi Square Test

Number of intervals = 7
Degrees of freedom = 4
Test Statistic = 12.5
Corresponding p-value = 0.0158

Kolmogorov-Smirnov Test

Test Statistic = 0.0674
Corresponding p-value > 0.15

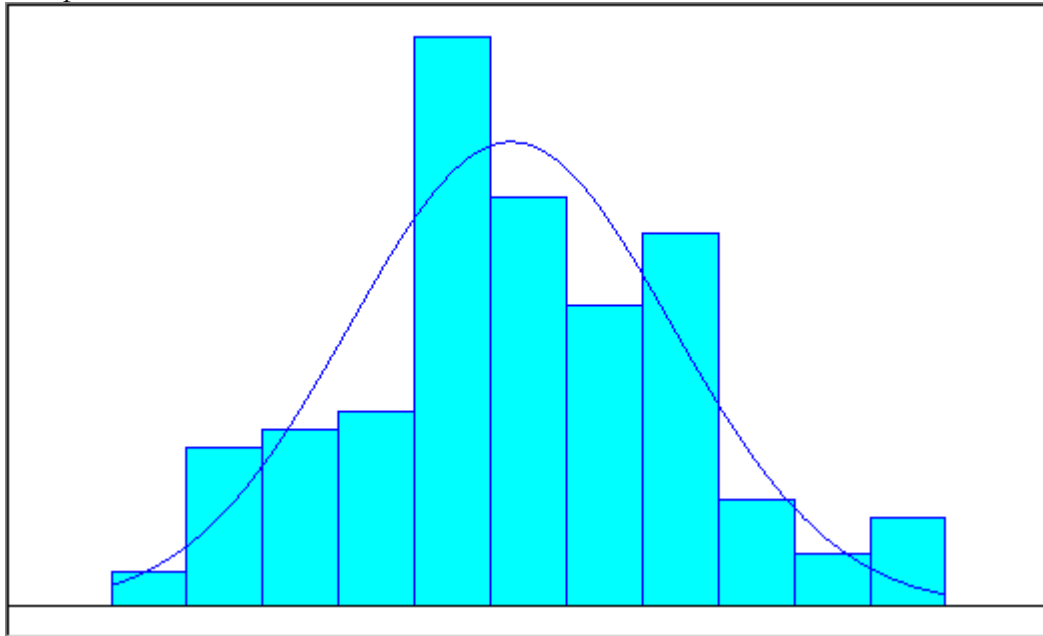
Data Summary

Number of Data Points = 185
Min Data Value = 8.42
Max Data Value = 11.8
Sample Mean = 10.1
Sample Std Dev = 0.649

Histogram Summary

Histogram Range = 8.07 to 12
Number of Intervals = 13

20 Rpm



Distribution Summary

Distribution: Normal
Expression: **NORM(20.8, 1.52)**
Square Error: 0.011151

Chi Square Test

Number of intervals = 7
Degrees of freedom = 4
Test Statistic = 10.2
Corresponding p-value = 0.039

Kolmogorov-Smirnov Test

Test Statistic = 0.0771
Corresponding p-value > 0.15

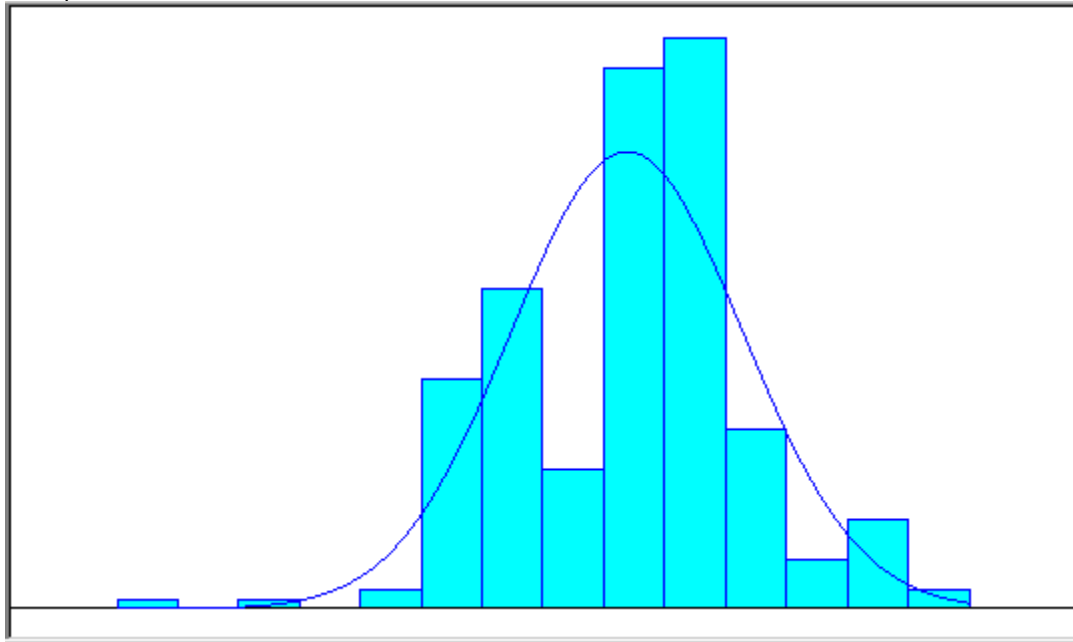
Data Summary

Number of Data Points = 139
Min Data Value = 17.1
Max Data Value = 24.2
Sample Mean = 20.8
Sample Std Dev = 1.52

Histogram Summary

Histogram Range = 17 to 24.9
Number of Intervals = 11

30 Rpm



Distribution Summary

Distribution: Normal
Expression: **NORM(30.2, 2.31)**
Square Error: 0.028861

Chi Square Test

Number of intervals = 7
Degrees of freedom = 4
Test Statistic = 36
Corresponding p-value < 0.005

Kolmogorov-Smirnov Test

Test Statistic = 0.0964
Corresponding p-value = 0.034

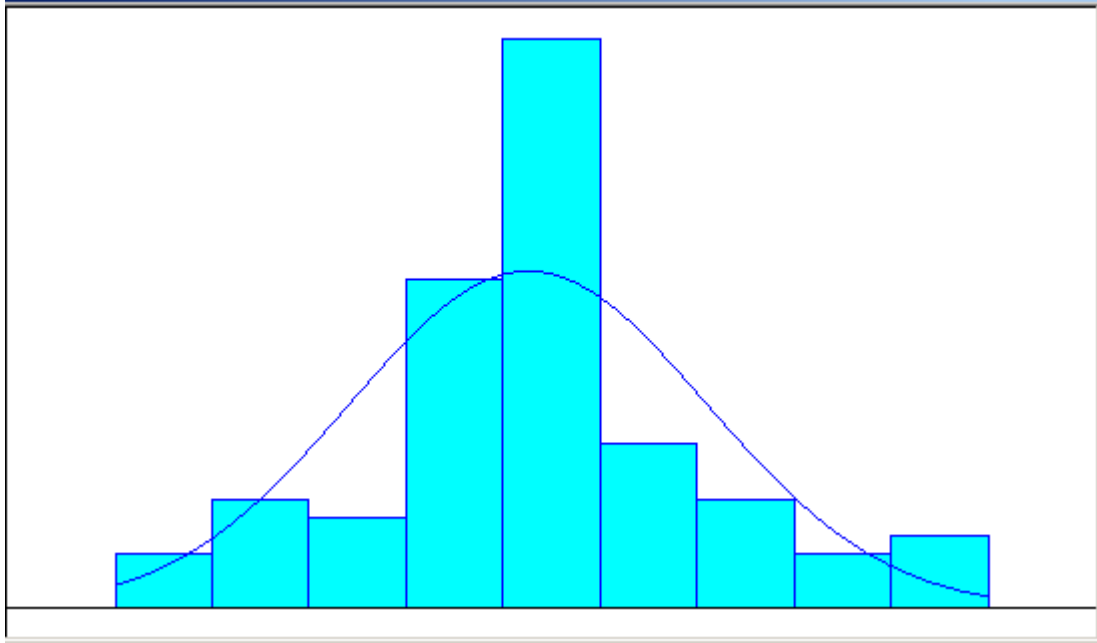
Data Summary

Number of Data Points = 218
Min Data Value = 20
Max Data Value = 36.6
Sample Mean = 30.2
Sample Std Dev = 2.31

Histogram Summary

Histogram Range = 20 to 37
Number of Intervals = 14

40 Rpm



Distribution Summary

Distribution: Normal
Expression: **NORM(40.5, 3.69)**
Square Error: 0.036211

Chi Square Test

Number of intervals = 6
Degrees of freedom = 3
Test Statistic = 16.4
Corresponding p-value < 0.005

Kolmogorov-Smirnov Test

Test Statistic = 0.143
Corresponding p-value = 0.0566

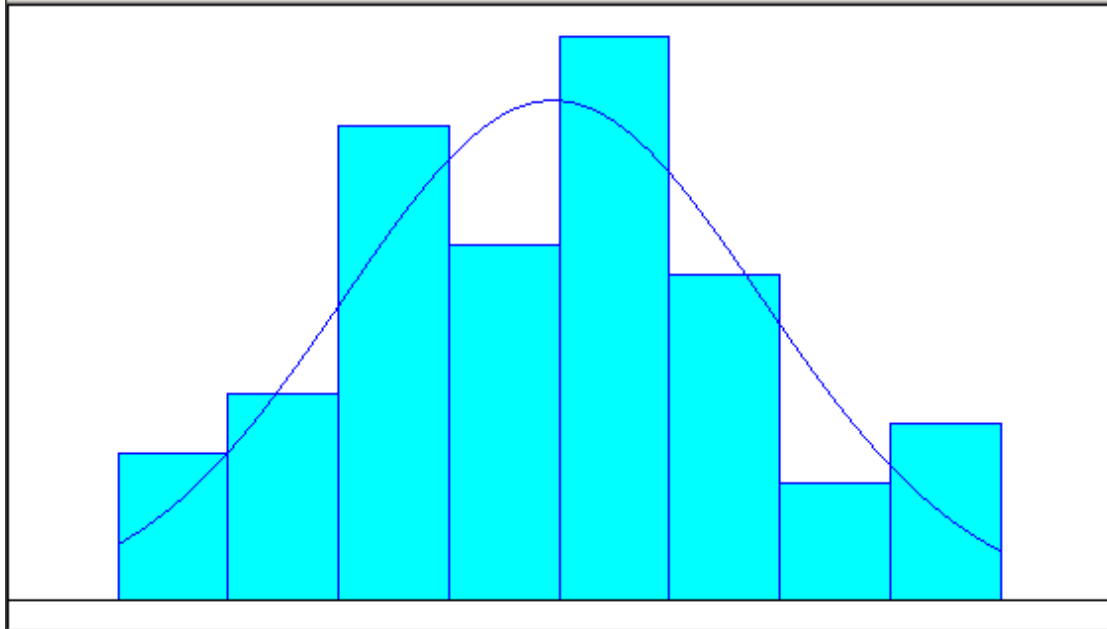
Data Summary

Number of Data Points = 85
Min Data Value = 32
Max Data Value = 49.3
Sample Mean = 40.5
Sample Std Dev = 3.71

Histogram Summary

Histogram Range = 32 to 50
Number of Intervals = 9

50 Rpm



Distribution Summary

Distribution: Normal
Expression: **NORM(49.9, 4.25)**
Square Error: 0.009379

Chi Square Test

Number of intervals = 6
Degrees of freedom = 3
Test Statistic = 2.85
Corresponding p-value = 0.43

Kolmogorov-Smirnov Test

Test Statistic = 0.119
Corresponding p-value > 0.15

Data Summary

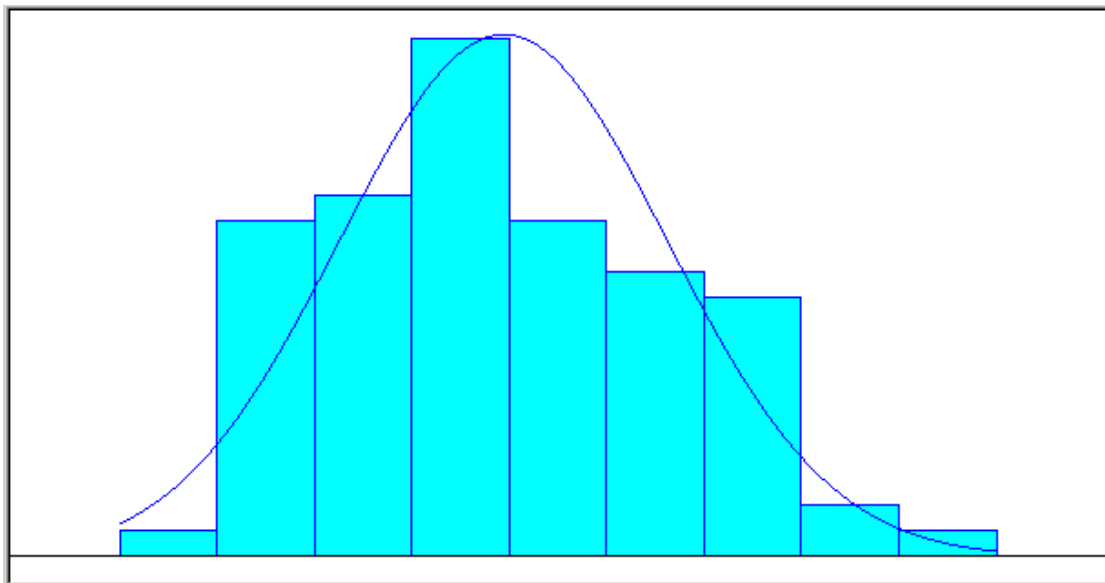
Number of Data Points = 80
Min Data Value = 41.2
Max Data Value = 58.1
Sample Mean = 49.9
Sample Std Dev = 4.28

Histogram Summary

Histogram Range = 41 to 59
Number of Intervals = 8

Program 2

10 Rpm



Distribution Summary

Distribution: Normal
Expression: **NORM(10.2, 0.714)**
Square Error: 0.012284

Chi Square Test

Number of intervals = 6
Degrees of freedom = 3
Test Statistic = 6.87
Corresponding p-value = 0.0801

Kolmogorov-Smirnov Test

Test Statistic = 0.0951
Corresponding p-value > 0.15

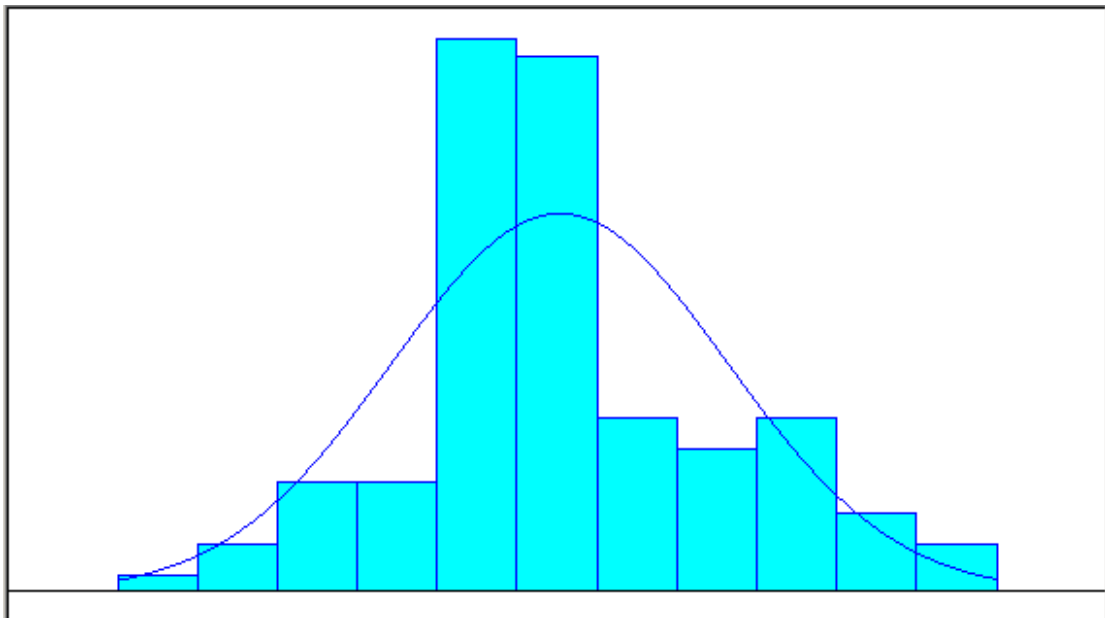
Data Summary

Number of Data Points = 85
Min Data Value = 8.89
Max Data Value = 12.1
Sample Mean = 10.2
Sample Std Dev = 0.719

Histogram Summary

Histogram Range = 8.57 to 12.4
Number of Intervals = 9

20 Rpm



Distribution Summary

Distribution: Normal
Expression: **NORM(21, 2.29)**
Square Error: 0.032806

Chi Square Test

Number of intervals = 7
Degrees of freedom = 4
Test Statistic = 28.3
Corresponding p-value < 0.005

Kolmogorov-Smirnov Test

Test Statistic = 0.153
Corresponding p-value < 0.01

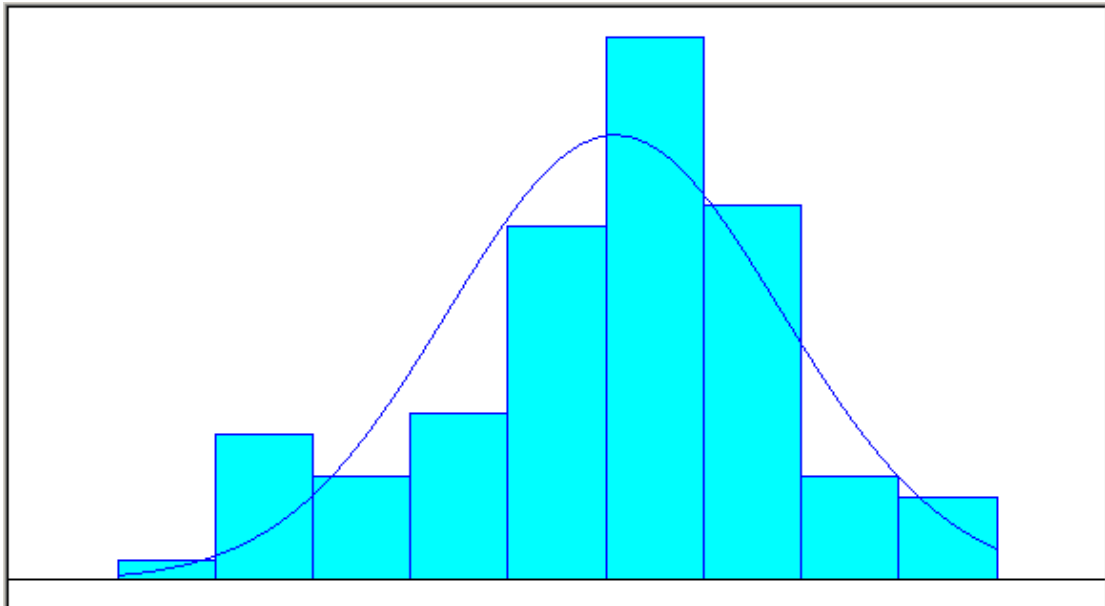
Data Summary

Number of Data Points = 126
Min Data Value = 15.6
Max Data Value = 26.9
Sample Mean = 21
Sample Std Dev = 2.29

Histogram Summary

Histogram Range = 15 to 27
Number of Intervals = 11

30 Rpm



Distribution Summary

Distribution: Normal
Expression: **NORM(29.8, 2.27)**
Square Error: 0.013942

Chi Square Test

Number of intervals = 6
Degrees of freedom = 3
Test Statistic = 6.24
Corresponding p-value = 0.101

Kolmogorov-Smirnov Test

Test Statistic = 0.113
Corresponding p-value > 0.15

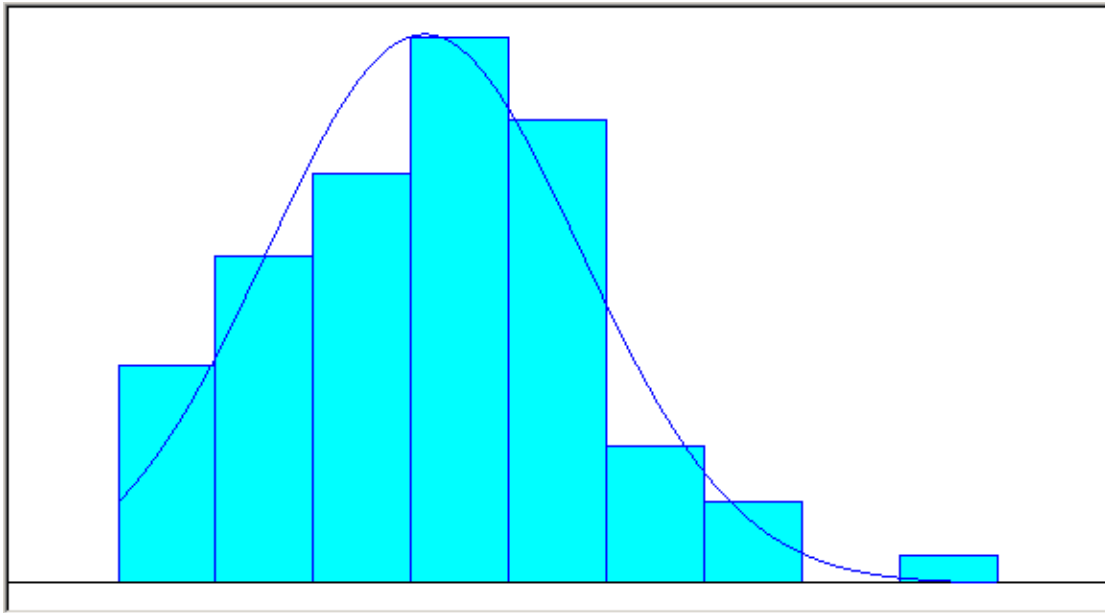
Data Summary

Number of Data Points = 91
Min Data Value = 23.1
Max Data Value = 34.4
Sample Mean = 29.8
Sample Std Dev = 2.28

Histogram Summary

Histogram Range = 23 to 35
Number of Intervals = 9

40 Rpm



Distribution Summary

Distribution: Normal
Expression: **NORM(40.9, 2.5)**
Square Error: 0.005094

Chi Square Test

Number of intervals = 5
Degrees of freedom = 2
Test Statistic = 1.95
Corresponding p-value = 0.399

Kolmogorov-Smirnov Test

Test Statistic = 0.0745
Corresponding p-value > 0.15

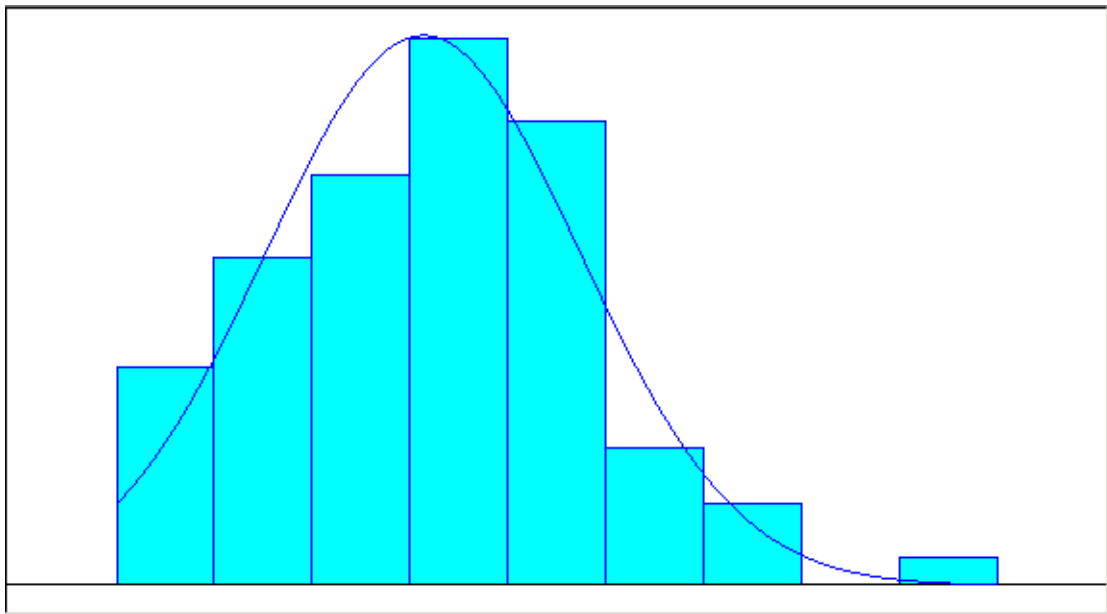
Data Summary

Number of Data Points = 81
Min Data Value = 36.1
Max Data Value = 49.3
Sample Mean = 40.9
Sample Std Dev = 2.51

Histogram Summary

Histogram Range = 36 to 50
Number of Intervals = 9

54 Rpm (This was actually measured at reference speed 54 Rpm, not 50 Rpm)



Distribution Summary

Distribution: Normal
Expression: **NORM(53.2, 3.08)**
Square Error: 0.025306

Chi Square Test

Number of intervals = 4
Degrees of freedom = 1
Test Statistic = 3.3
Corresponding p-value = 0.074

Kolmogorov-Smirnov Test

Test Statistic = 0.116
Corresponding p-value > 0.15

Data Summary

Number of Data Points = 37
Min Data Value = 45.7
Max Data Value = 60
Sample Mean = 53.2
Sample Std Dev = 3.12

Histogram Summary

Histogram Range = 45 to 60
Number of Intervals = 6

APPENDIX C: FULL LIST OF DATA COLLECTED

Method 1

10 RPM

10.157	10.411	9.770	10.320	10.081	9.926	9.921	11.534	9.921	9.847
9.847	11.527	9.921	10.576	9.625	11.031	9.625	10.241	10.406	9.625
10.667	9.926	10.667	11.848	11.130	9.921	10.081	10.000	9.695	9.770
10.846	9.625	9.625	9.770	9.699	9.926	10.081	11.038	9.921	9.141
10.235	9.695	9.412	10.241	10.241	10.076	10.320	9.620	10.411	9.847
9.921	9.770	10.846	11.230	8.478	10.241	10.076	10.406	10.667	10.582
10.081	10.320	10.325	10.667	10.406	10.493	10.076	10.320	10.576	10.667
10.941	10.411	10.081	9.699	10.000	10.081	11.429	10.157	11.130	9.625
9.770	11.230	9.926	10.320	10.241	10.081	10.935	10.758	10.241	9.625
10.576	11.325	11.031	9.695	9.921	9.346	8.421	10.667	9.208	9.208
8.826	8.707	10.753	11.325	10.935	8.949	10.406	9.272	10.406	9.699
9.346	9.770	9.620	9.770	10.758	9.556	9.770	10.406	10.488	9.625
9.079	9.847	10.582	9.074	8.889	9.208	8.647	8.707	9.208	9.479
9.412	10.241	9.847	9.479	9.551	9.074	9.276	9.346	10.241	9.770
9.208	9.551	9.625	9.625	9.625	9.770	9.620	11.130	10.582	9.013
9.921	10.163	10.241	9.921	9.551	9.625	9.921	9.770	11.527	10.000
11.429	9.921	9.625	9.695	10.081	10.000	10.000	9.926	9.926	9.921
9.921	10.758	10.846	11.325	9.272	10.081	9.770	10.406	9.921	10.235
11.635	9.921	9.483	10.081	10.320					

20 RPM

20.305	23.697	18.832	22.051	21.692	20.000	21.322	20.640	20.000	18.815
19.704	20.000	22.075	19.102	20.325	19.685	21.345	18.282	22.075	20.325
20.000	20.325	20.000	20.640	18.034	20.640	19.685	21.322	20.000	20.325
19.102	18.298	23.725	17.286	19.102	19.399	17.778	20.000	17.065	23.283
18.034	20.000	18.282	20.000	19.704	18.553	20.325	20.305	24.155	21.322
21.345	20.305	21.345	20.986	21.322	22.472	20.986	24.155	20.000	21.322
22.075	22.857	18.298	19.399	21.692	22.051	22.075	20.986	20.661	19.380
22.447	21.692	20.305	20.986	19.120	22.472	21.322	20.640	20.000	22.075
20.000	21.322	20.000	20.986	20.986	21.345	22.075	20.000	22.075	20.325
22.447	22.051	22.075	18.832	20.640	18.815	20.986	24.155	20.964	20.986
19.685	18.034	20.640	20.986	20.325	20.000	18.018	20.325	20.986	20.661
19.685	20.986	20.000	18.815	22.075	23.283	22.857	20.000	22.472	21.345
20.964	22.447	23.697	22.075	23.283	24.155	24.155	22.075	22.857	20.986
21.345	19.685	20.325	20.640	19.685	20.000	20.000	21.322	22.447	

30 RPM

29.762	29.762	27.248	32.841	31.201	30.488	31.250	28.449	31.201	32.787
29.762	27.211	27.816	29.762	28.449	30.441	29.762	31.201	31.250	32.000
32.841	27.816	31.201	32.841	31.250	31.201	28.409	32.841	28.409	30.441
34.602	29.762	28.409	31.201	28.409	29.762	27.211	27.211	31.250	29.112
27.816	20.000	29.806	29.762	27.248	28.449	31.250	29.762	31.201	28.449

29.762	29.762	32.841	31.201	31.201	29.762	31.201	27.248	32.000	28.450
31.201	26.110	31.250	32.841	31.201	29.112	28.449	33.670	34.542	29.070
34.602	29.112	29.762	29.112	27.855	27.816	29.070	26.110	33.670	32.000
32.000	32.841	29.762	31.201	31.201	26.667	31.201	29.112	28.449	31.250
31.201	30.488	33.670	28.449	30.441	28.449	29.762	32.841	31.250	29.806
30.488	32.841	27.248	32.000	31.201	32.000	29.070	33.727	36.563	31.250
32.841	28.449	28.450	30.441	32.000	31.250	29.112	29.070	27.248	28.449
32.787	35.524	28.450	29.070	26.144	31.201	25.575	31.201	31.201	30.488
32.841	26.144	29.762	36.563	32.841	29.762	29.762	31.201	31.250	29.762
35.587	31.201	29.762	26.110	27.248	29.762	29.762	29.762	27.211	31.201
30.488	28.409	29.762	29.762	34.602	27.816	32.787	29.806	27.248	31.250
27.248	32.000	29.762	28.450	34.602	29.762	32.841	27.816	31.201	29.762
29.762	26.144	29.762	32.000	29.762	29.070	27.248	31.201	29.762	29.762
23.283	29.762	31.201	27.855	29.762	29.806	29.762	27.248	31.250	31.250

40 RPM

42.644	42.644	38.835	34.602	34.602	40.000	42.644	35.524	40.000	49.261
49.140	38.760	47.506	41.322	41.237	38.835	38.835	38.835	38.760	40.000
40.000	40.000	38.760	42.735	38.760	38.835	37.665	38.835	42.644	41.322
41.322	42.644	38.760	41.322	38.835	41.237	42.644	40.000	34.602	36.563
38.760	41.322	41.237	40.000	34.602	40.000	41.322	40.000	32.000	32.000
40.000	36.563	38.760	41.237	38.835	49.261	41.237	38.760	41.322	41.322
38.760	41.322	40.000	40.000	36.563	35.524	40.000	45.767	40.000	36.563

44.150	47.393	38.760	42.735	49.261	41.322	41.237	44.150	41.322	45.662
47.393	42.644	45.662	44.150	32.000					

50 RPM

51.151	58.140	44.150	53.333	58.139	53.333	47.393	51.282	51.282	51.151
45.767	58.139	51.282	49.140	53.333	58.140	47.393	49.261	51.151	49.261
49.261	53.333	55.710	55.710	49.261	55.710	44.150	49.261	49.261	51.282
41.322	44.150	51.282	53.333	47.393	51.151	42.644	49.261	44.150	51.282
51.151	47.393	49.261	47.393	47.393	45.662	47.506	58.140	53.333	51.282
49.140	47.393	45.662	51.151	51.151	41.322	51.151	58.140	49.261	44.053
53.333	49.140	55.710	44.150	41.237	53.333	47.393	53.333	53.333	44.150
51.282	51.151	47.393	51.151	41.322	45.767	53.333	45.662	47.393	51.282
51.151	58.140	44.150	53.333	58.139	53.333	47.393	51.282	51.282	51.151
45.767	58.139	51.282	49.140	53.333	58.140	47.393	49.261	51.151	49.261
49.261	53.333	55.710	55.710	49.261	55.710	44.150	49.261	49.261	51.282
41.322	44.150	51.282	53.333	47.393	51.151	42.644	49.261	44.150	51.282
51.151	47.393	49.261	47.393	47.393	45.662	47.506	58.140	53.333	51.282
49.140	47.393	45.662	51.151	51.151	41.322	51.151	58.140	49.261	44.053
53.333	49.140	55.710	44.150	41.237	53.333	47.393	53.333	53.333	44.150
51.282	51.151	47.393	51.151	41.322	45.767	53.333	45.662	47.393	51.282
51.151	58.140	44.150	53.333	58.139	53.333	47.393	51.282	51.282	51.151
45.767	58.139	51.282	49.140	53.333	58.140	47.393	49.261	51.151	49.261
49.261	53.333	55.710	55.710	49.261	55.710	44.150	49.261	49.261	51.282

Method 2

10 RPM

10.406	9.699	9.690	10.320	10.846	11.707	9.625	9.843	9.407	10.152
11.038	10.081	10.846	9.276	11.223	10.331	11.236	10.320	10.163	9.416
10.493	9.852	10.157	10.163	10.163	9.843	10.320	9.699	10.000	10.406
10.846	10.846	10.493	10.000	9.017	10.241	10.163	9.770	10.406	9.407
9.079	9.276	9.699	10.000	9.843	10.152	11.223	10.000	9.141	9.843
9.407	9.407	10.000	9.551	11.236	9.551	8.889	9.017	11.295	11.038
11.641	9.416	10.000	9.141	9.551	11.176	11.429	11.429	10.163	11.236
10.331	11.038	10.672	10.000	9.770	12.077	11.223	10.163	11.025	10.163
10.858	10.846	11.038	10.320	10.582					

20 RPM

20.216	19.595	18.315	19.550	20.975	19.380	21.517	19.659	19.516	22.075
16.863	22.272	20.647	23.708	17.301	21.042	19.695	22.650	17.301	24.845
23.708	24.631	25.856	26.441	16.835	25.813	20.422	24.783	17.301	20.661
17.301	20.640	19.417	26.667	22.075	21.716	20.661	22.075	20.000	20.471
20.000	18.553	19.851	20.481	24.570	20.211	20.661	20.325	20.986	19.417
17.762	17.301	20.619	22.075	15.625	25.575	20.305	23.697	18.315	19.846
24.631	20.161	19.120	22.075	23.753	19.380	19.380	21.413	20.661	20.000
20.619	20.640	17.301	24.426	22.070	18.282	26.896	22.857	23.697	20.986

20.964	22.059	22.302	16.420	20.647	22.861	21.322	21.322	20.812	22.447
24.618	24.570	19.380	20.000	22.831	20.661	20.422	23.810	20.812	20.661
20.000	20.000	20.964	23.817	20.986	19.704	21.368	25.202	20.862	19.380
23.697	20.000	24.610	20.661	20.000	20.000	19.508	20.812	19.120	19.102
19.595	19.380	20.795	20.251	20.661	20.986				

30 RPM

30.722	29.098	27.855	28.409	29.091	29.762	28.249	28.978	30.488	26.667
27.248	30.667	31.907	32.000	33.831	30.597	32.017	32.841	31.825	29.904
30.000	29.070	30.488	29.070	30.488	27.447	30.722	32.450	31.368	32.051
30.968	29.433	29.311	30.000	31.447	29.528	31.596	27.248	32.841	25.641
31.250	30.395	30.769	32.841	27.211	29.155	25.641	25.575	26.667	32.000
25.575	32.841	31.480	33.784	30.000	29.771	30.722	33.727	27.248	25.575
23.148	29.904	31.949	26.667	31.949	29.542	24.631	29.070	30.769	31.082
30.488	31.334	34.364	30.960	26.667	29.762	29.597	29.762	29.155	30.960
27.778	32.000	28.409	30.395	31.596	25.575	29.693	29.474	26.667	31.596
29.505									

40 RPM

41.322	42.438	40.989	45.662	40.872	41.017	39.159	37.106	40.872	39.262
42.553	41.017	42.657	44.150	42.735	39.533	42.644	37.895	36.232	36.843

37.657	37.594	36.563	38.835	36.946	40.630	37.629	39.735	40.000	42.674
41.379	40.404	37.641	36.052	40.333	49.261	42.674	42.353	44.150	37.665
41.456	42.735	38.835	40.404	40.184	41.866	39.841	40.369	40.733	41.973
41.053	40.844	41.168	42.735	38.610	40.587	45.872	40.766	40.000	39.769
41.322	42.614	44.643	42.644	37.250	44.150	38.760	42.662	43.134	41.322
42.553	41.725	42.735	42.123	42.689	41.322	44.150	45.767	39.062	37.641
37.027									

54 RPM

53.191	55.710	51.903	45.662	49.261	53.333	54.496	54.123	51.508	60.000
53.476	54.662	52.996	52.219	55.127	57.143	54.466	54.845	56.324	53.937
52.788	50.000	58.140	53.362	54.230	53.333	55.659	54.103	51.216	54.845
55.710	56.899	49.489	47.755	50.505	51.151	45.662			

APPENDIX D: ATTACHED CD CONTENTS

%CD%\Report\Aug07_SE490_Report_FanSpd.pdf

%CD%\Report\Aug07_SE490_Report_FanSpd.doc

%CD%\Aug07_SE490_Presentation_FanSpd.ppt

%CD%\Software\SE490_FanSpd_Final.vi

%CD%\Photos\

%CD%\Videos\