Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X
Volume 3, Issue 5, 2025

ANALYSIS OF AIRFOIL USING WIND TUNNEL THROUGH IMAGE PROCESSING

Abdul Sami*1, Arish Khan2, M. Uzair Taqi3, Syed Najeeb Haider Jafri4

*1,2,3,4 *College of Engineering, Karachi Institute of Economics and Technology (KIET) Karachi, Pakistan*

*1 abdulsamisoomro37@gmail.com, 2arishk103@gmail.com, 3uzairtaqi55@gmail.com, 4najeebhjafri_14@gmail.com

DOI: https://doi.org/10.5281/zenodo.15462737

Keywords
Aerodynamic parameters, Computational Fluid Dynamics (CFD), pressure sensors, Image Processing, CFD Simulations

Article History
Received on 08 April 2025
Accepted on 08 May 2025
Published on 17 May 2025

Corresponding Author: * Abdul Sami

Abstract
This project introduces a novel approach to measuring aerodynamic parameters using digital image processing, offering a simpler and more cost-effective alternative to traditional wind tunnel methods involving manometers and anemometers. The study explores three main methodologies, including Computational Fluid Dynamics (CFD), pressure sensors, and image processing, with a focus on the latter. By recording airflow behavior over an airfoil using a high-quality digital camera and analyzing the footage in MATLAB, pressure and velocity were determined effectively. A smoke generator was used to visualize airflow patterns in the wind tunnel. The image processing results showed strong agreement with anemometer readings and CFD simulations, confirming its reliability. This method simplifies aerodynamic testing and offers a more efficient alternative to conventional techniques.

1. INTRODUCTION

This project aims to develop a cost-effective and accurate method for analyzing the aerodynamic behavior of an airfoil using Digital Image Processing (DIP). Traditional methods such as manometers and anemometers, though effective, are complex and labor-intensive. By employing DIP with MATLAB software, along with tools like SolidWorks, STAR-CCM+, and a smoke generator, we seek to measure velocity and dynamic pressure over an airfoil placed in a subsonic wind tunnel[1].

The experimental setup includes a smoke generator, digital camera, wind tunnel, and a custom-designed airfoil. The smoke is introduced into the settling chamber of the wind tunnel, where it passes through a honeycomb structure to ensure laminar flow[2]. As smoke flows over the airfoil, a high-resolution video is captured using a digital camera. This video is then processed using MATLAB to apply optical flow algorithms, allowing visualization and calculation of smoke particle movement, ultimately yielding pressure and velocity values.

To validate the results, readings from an anemometer [3], and simulations from STAR-CCM+ are compared with DIP outputs. The airfoil used is modeled in SolidWorks and analyzed further using XFLR software for aerodynamic properties such as lift and stability.

Our primary objective is to determine the velocity and dynamic pressure of fluid (smoke) flow over an airfoil using a subsonic open-circuit wind tunnel located in the Propulsion Lab. The innovative element is the use of Digital Image Processing, which allows us to extract quantitative aerodynamic data from video footage, making the process more accessible and economical.

We designed the airfoil using SolidWorks and XFLR, and visualized the flow using a smoke generator. The airflow is recorded via a digital camera, and optical flow algorithms are applied in MATLAB to extract motion vectors from the smoke. These vectors are then used to calculate velocity and pressure values. Compared to traditional methods, DIP provides a computational yet reliable technique for aerodynamic analysis.

2. Key Components and Methodology

2.1. Wind Tunnel

A subsonic wind tunnel (Mach < 1) is used to simulate airflow over the airfoil. It allows control over flow conditions and provides a physical environment for DIP-based measurement and analysis.

2.2. Digital Camera

A high-definition digital camera captures video of the smoke flow. Adjustable parameters like resolution and frame rate (fps) play a critical role, as DIP algorithms rely heavily on pixel intensity and clarity to track motion accurately.

2.3. Smoke Generator

Since air is transparent, **smoke** is used to make flow patterns visible. This helps DIP algorithms identify motion across frames. The smoke generator used can produce flow speeds up to 30 m/s.

2.4. Software Tools

2.4.1. MATLAB: Core tool for implementing DIP algorithms and analyzing motion data.

2.4.2. SolidWorks & XFLR: Used to design and analyze the airfoil model.

2.4.3. STAR-CCM+: Used for CFD simulations and comparison with experimental results.

2.5. Design Objectives and Focus

The design objective is to accurately measure velocity and dynamic pressure of smoke over an airfoil using DIP. This is essential because these two parameters serve as the foundation for calculating other aerodynamic forces such as lift and drag. Key goals include:

1. Visualize and analyze airflow over the airfoil using smoke.
2. Extract and compute velocity and pressure data using optical flow algorithms.
3. Validate results with anemometer readings and CFD simulation data.

2.6. Challenges and Analysis

2.6.1. Design Issues

2.6.1.1. Smoke velocity: Limited to ~30 m/s due to generator constraints.

2.6.1.2. Processor limitations: DIP performance is heavily dependent on high processing power (≥3.0 GHz recommended).

2.6.1.3. Video resolution mismatch: Affects accuracy if the recorded resolution and algorithm input do not align.

2.6.2. Design Analysis

2.6.2.1. Video resolution and its compatibility with processor capability.

2.6.2.2. Frame rate consistency, as mismatched fps affects algorithm output.

2.6.2.3. Flow resolution to avoid noise in DIP results.

2.6.2.4. Code-video dimension match to ensure correct scaling of optical flow vectors.

2.6.2.5. Optical flow

Optical flow defines the movements of the something picked from a captured video. It reads and proceed the data frame by frame in a video and observes only the depiction

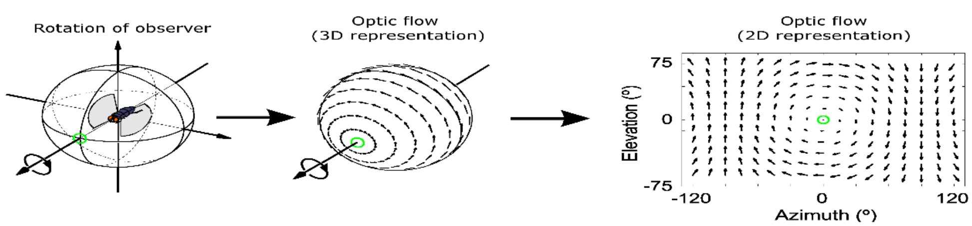

Figure 01 2D optical flow Representation

of the motion caused. Optical [4] can be tracked in both of the videos either recorded in 2D or in 3D. [Figure 01] represents the 2D and 3D motion. The 2D video motion variable is I with respect to (x,y) according to the ---- where I is the gradient pixel of the video with respect to X and Y. the velocities of the particles would be represented as Vx and Vy.

The optical flow analysis gives the difference over the frames as soon as the frames continuosly moves.

$$I_x = \frac{\partial I}{\partial x}, I_y = \frac{\partial I}{\partial y}$$

Where the Ix defines the pixel intensity in relation with x component and Iy defines the pixel intensity in relation with y component as in 2D.

Vx it is the velocity of the X component

Vy it is the velocity of the Y component

On the other side the 3D optical flow analysis includes the I (X , Y, Z). these are the derivatives of the pixel intenisty gradient. They are also known as the partial derivatives.

The 3D differs in axis components as 3D is in 3 dimension so the z axis component will also be included represented as.

$$I_xV_x + I_yV_y + I_zV_z$$

2D videos interpret the motion in 2 dimensional space while the 3D videos created the motion in 3 dimensional space. Optical flow estimation consist of some methods that are used to define it. however the methods are given below[5].

2.6.2.5.1. Gradient methodology

Gradient method reads the iterations in pixel by pixel. This method includes the two techniques as defined lucas and kannade technique and S Horn chunk technique. These two methods detect the movement of the object act like sensor. It read the pixels of video, depict and track the motion particles. It operates by using the partial derivatives.

2.6.2.5.2. Correlation methodology

The video comprised of the number of frames by finding a window. Each frame distributed and get into the block and when they matched this defines the block matching method. This technique resembles with the block matching technique of optical flow analysis. This method denotes the two levels one is highest level means high resolution and other one is low resolution as lowest level. In this method the process begin from low resolution pixels. Every next frame assumes previous frame as a reference as shown in [Figure 02].

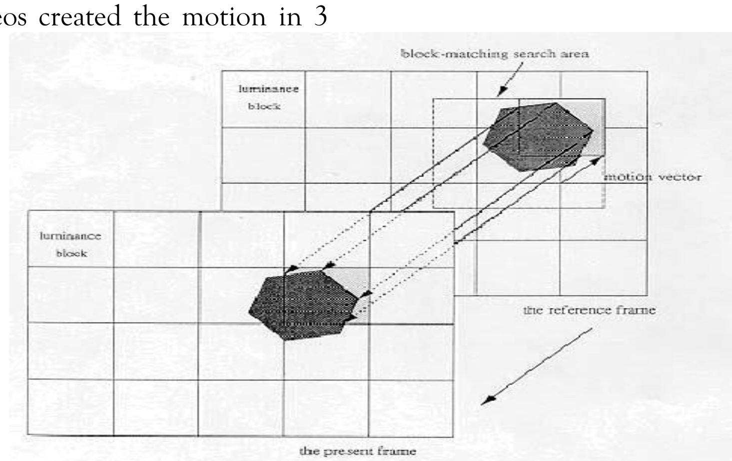

Figure 02 Block matching technique

2.6.2.5.3. Spatial methodology

This method depends on the frequencies of the pixel intensity which denotes the speed of the moving object. The movements happen due the variation in frequencies known as the spatial frequencies. This method uses the spatio filters. It is accurate but depends on the pixel energies.

2.7. Gradient Technique Approach

The gradient technique consist of two sub methods. One is lucas kannade and the other one is the horn s

ISSN (e) 3007-3138 (p) 3007-312X Volume 3, Issue 5, 2025

chunk method as shown in the [Figure 03]. It computes the intensities of pixels in an image [i].

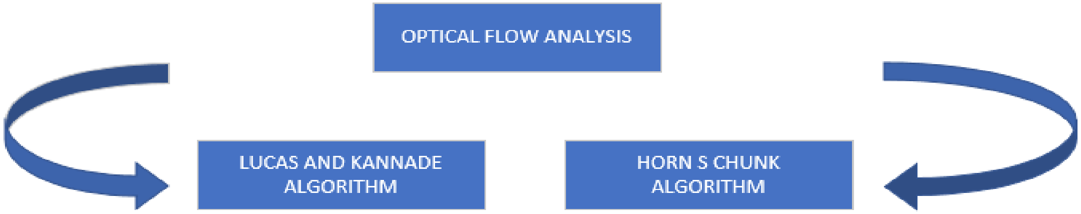

Figure 03 Optical flow flowchart

2.7.1. Lucas Kannade Method

The mostly used method is the lucas kannade method in optical flow analysis as a differential method . This technique based on the optical flow equation and solves it. this solves the equation but not give the flow data in the inner of constant regions. This method suppose that the displacement between the consequent frames are very small. It is very slow method. This algorithm determine the positions of vectors although it also tracks the vectors of the flow particles. Hence it is also modified.

This computational algorithm also have another feature to depict the edges the image that provides significant data. It also removes the blockage and prevent the unstability of the vectors detection.

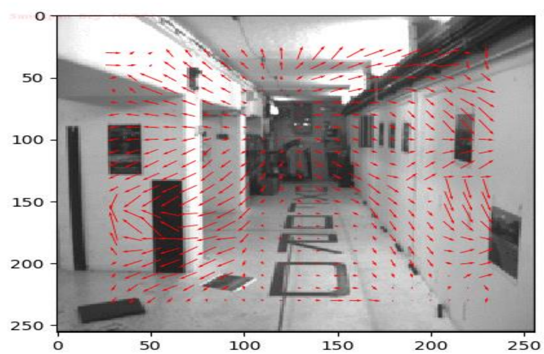

Figure 04 LK algorithm output

Figure 05 Sample video in vector form

2.7.2. Output representation of the LK

The LK algorithm technique represent its output in the form of the vectors detected from the smoke flow over any object. As it provides the magnitude of the vector and their direction with an arrow head. It predicts the uniform flow. It used the derivatives of X and Y in two dimensions. It uses the least square method.

Figure 05 LK output representation

[Figure 05] shows the green vectors with their magnitude and direction. This video is recorded by us for the initial stage practice. However, the results are shown in [Figure 05] with the accuracy.

2.7.3. LK working Strategy

It needs some steps to complete the processing on the video. First of all it detects all the pixels present across the each frames present in the video then it calculates their intensities. It will make the gradient matrix and pickup their eigen values. It stores the values of the pixel positions all across the frames and it detects the high pixel and use them to track then it depict the velocities and uses the gaussian filter in order to reduce the noise and it decides each pixels according to their weightage and shows their results in vector form as shown in [Figure 6]. It gives the green vectors in good estimation and shows vectors in red colour when there in bad estimation[6].

Figure 06 Sample video representation of LK

2.7.4. LK method

Lucas kanndae method is the most common and efficient method which is also used mostly in the flow analysis era because it provides the results in the form of moving vectors. It is not very complex as compared to others. It computes with the linearity calculation. It maintains the stability of the video in terms of frame size.

2.8. Horn S Chunk Method

Horn S Chunk method [7] is another technique that can give optical flow analysis. However, it is different from the Lucas and Kannada methods but has the same phenomenon. The Lucas Kannada method also has a problem of brightness and shading. To overcome this, the horn's chunk method is used. To reduce this problem, a gradient-based method is also used.

u = dx/dt v = dy/dt

It uses the U and V that are the velocities of the flow particles. It uses the U and V laplacian. It operate the algorithm in iterative form. It comes to overcome the lucas kannade by removing the aperture problem and gives the stability and smoothness. It also have removed the noise and distortion as detected. It uses the colours to represent the flow. The flow levelness and iterations are very important factors of this technique. The iteration area only notify the redundancy of the loop simulation.

2.8.1. Working of HS technique

HS method works in a different manner as compared to LK. Whereas the strategy would be the same. It gives the brightness constancy and it take the leveled brightness with respect to the neighbouring pixels in each frames. This method uses the partial derivative as Ix depend on x axis, Iy that depends upon y axis and It that depends on time. It solves the gauss siedal equation for the motion velocity estimation. This method stop the iterations till the last frame come.

2.8.2. Output representation of the HS

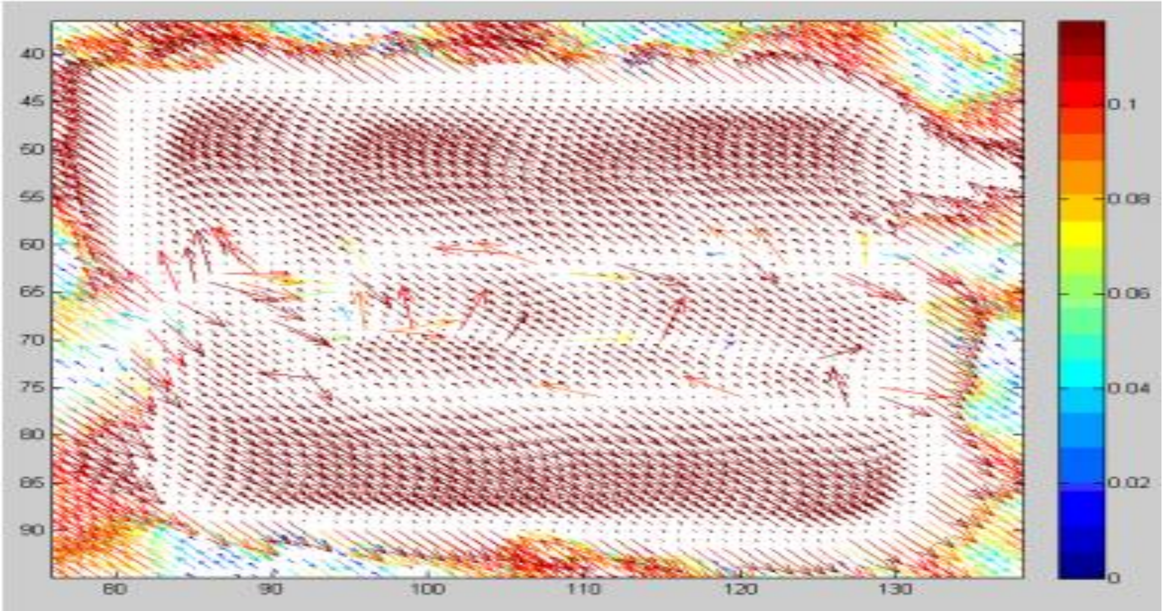

Figure 07 HS output representation

[Figure 07] gives the output not in vector form infact it gives the results in the colours coded demonstration according to their intensities at different points and also shows the vectors in colour form by using the quiver plot that gives the vector demonstration. On the right side it also provides the colour bar that denotes the strength and inensities of each pixel.

2.8.3. Why using HS

HS algorithm is same as LK but it removes those errors that LK had been facing. It means it is updated LK but we use it for accuracy as well as it takes less time to run as compared to LK and no noise at edge detection and takes both low and high intensities pixels that makes the solid output.

3. HARDWARE IMPLEMENTATION

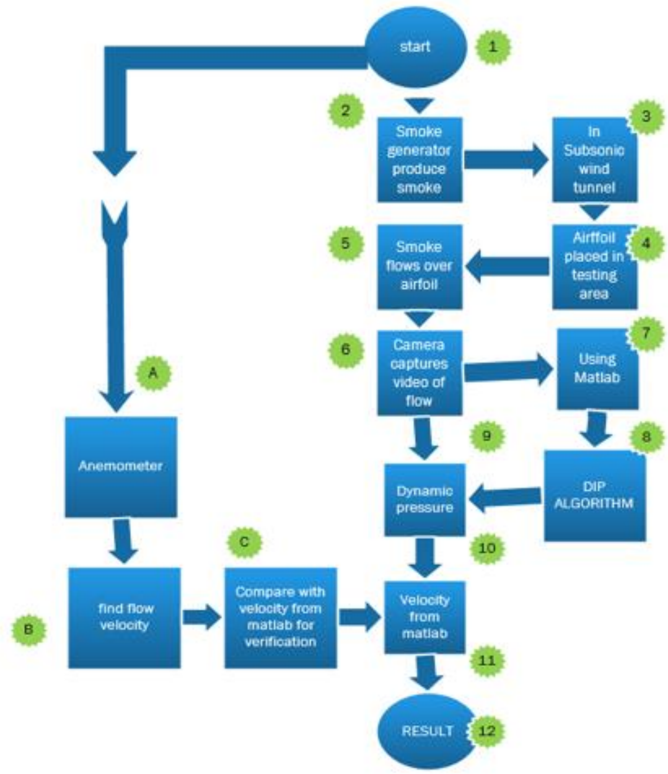

Figure 08 Project Hardware overall diagram

3.1. Designing an Airfoil 4415

Designing an airfoil is the main purpose for this project because airfoil is the testing object otherwise you can use another object to be tested. We selected cambered airfoil which is 4415.

3.1.1. Solidwork

The solidwork is used here to design airfoil as in [Figure 09]. We first calculated its parameters and selected the configuration data as 4415 and designed by using CAD software solidwork. The other objects can be used inorder to know the orientation. However the NACA airfoil is commonly used in aviation.

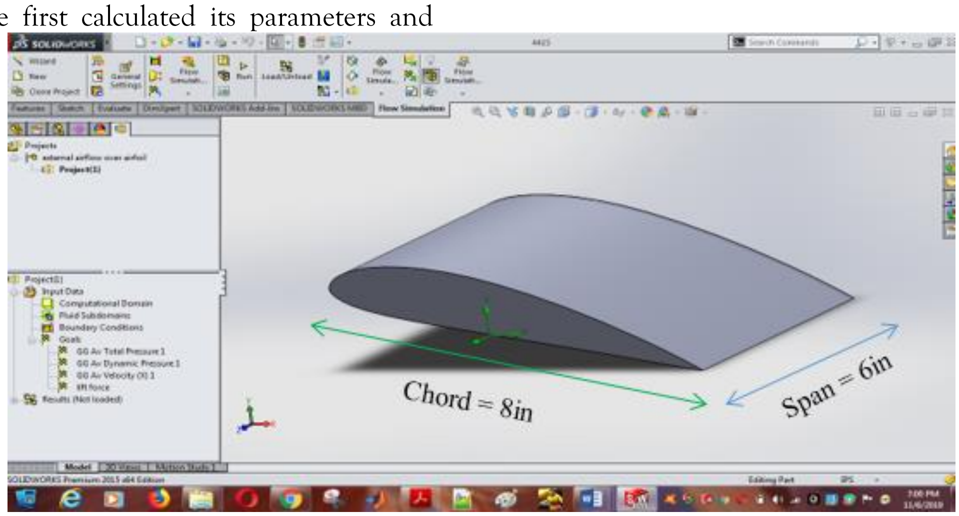

Figure 09 Design of airfoil 4415

3.2.1 Implementation 4415 as Hardware

The balsa and ply wood used to design this airfoil after designing in solidworks. We used cambered airfoil to get to know the analysis of the flow as in [Figure 10].

Figure 10 Prototype hardware of 4415 cambered airfoil

3.2.2. Digital Camera

The digital camera [Figure 11(a,b)] is used to captured the video of the flow over the airfoil in which the DIP algorithm have to apply. Remember one thing camera should have good resolution and features in order to change their frame rate and pixels. The better FPS is 60.

Figure 11(a,b) Digital Camera front view and rear view

3.3 Smoke Generator

Smoke generator as in [Figure 12] used to generate smoke our objective is to analyze the flow over the airfoil 4415. Smoke generator is placed in the settling chamber of the windtunnel.

Figure 12 Smoke generator

3.4 Digital Anemometer

We are using anemometer [8] as in [Figure 13] in this project to verify the predicted velocity of the flow over the airfoil. Predicted velocity is that velocity that is predicted from the lucas kannade and Horn s chunk algorithms.

Figure 13 Digital anemometer

3.5 Initial Practice Videos

Before going towards the final results we worked on some samples of captured video for initial practice testing in order to know the action of the flow. We used the lucas kannade and Horn chunk technique as [Figure 14-16].

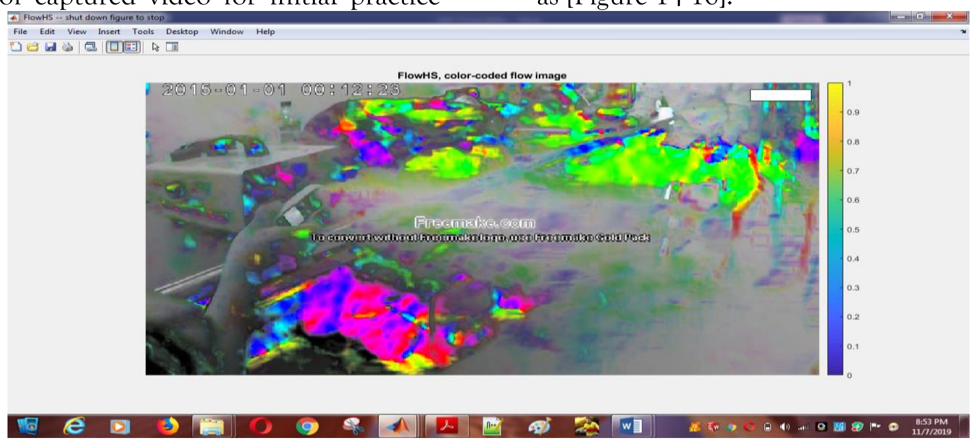

Figure 14 Initial Captured sample video on colour coded

Figure 15 Optical Flow in HS colour coded

ISSN (e) 3007-3138 (p) 3007-312X Volume 3, Issue 5, 2025

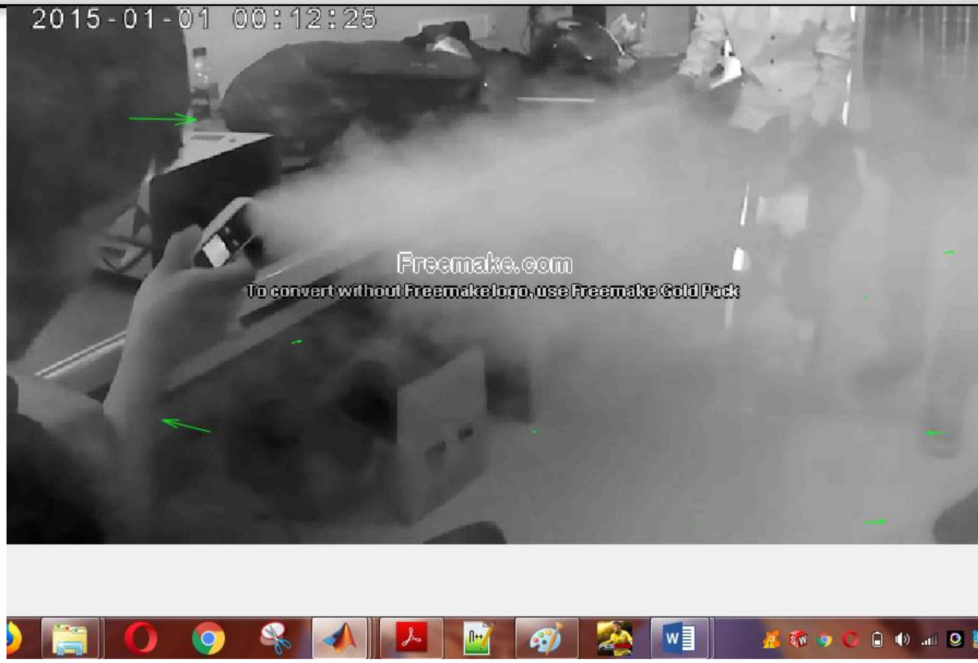

Figure 16 Optical Flow output in LK Representation

3.5.1 Block Diagram of working

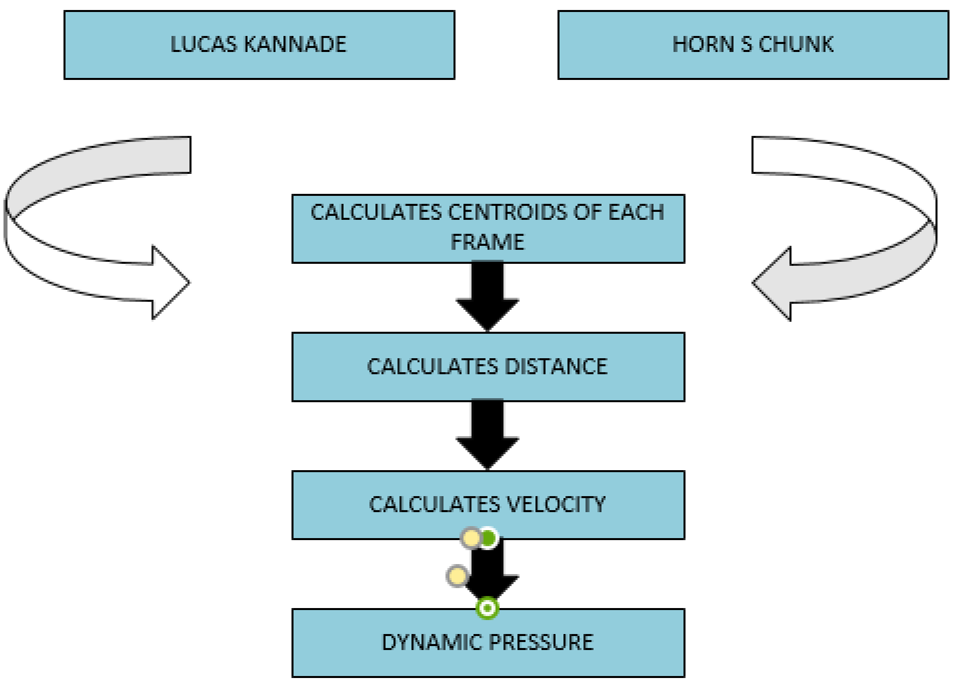

Figure 17 LK and HS Working strategy

3.5.2 Distance

Before predicting the flow velocity it is necessary to find the distance each distance of the flow particles. Xi and Yi is the distance in x direction where Xj and Yj is distance in y direction. Distance between i and j is distance between two centroids of the frames.

Centroid is the center or pixel average of the frames of video [9]

$$dist_{ij} = \sqrt{(x_i-x_j)^2 + (y_i-y_j)^2}.$$

This is also known as euclidean formula.

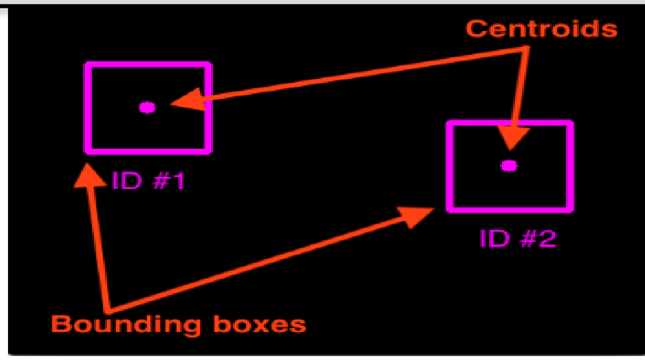

Figure 18 Calculating the Centroids of Frames

3.5.3 Time
Time in image processing is calculated when there is inverse relation of frame rate.
Time = 1/frame rate

3.5.4 Velocity
As every one knows the formula of the velocity that must used in this project inorder to get the velocity of the flow in DIP. The formula for velocity is
Velocity = distance / time
Distance as calculated divide by the time and measured in meter per second.

3.5.5 Dynamic pressure
Our second aim is to depict the dynamic pressure [ii] whose formula is

$q = ½ \rho v^2$

Where q = dynamic pressure
ρ = density of the fluid
V = velocity of fluid

3.6 Optical flow LK Algorithm
The lucas kannade is first calculates the pixel strength and then solves the gradient matrix and their eigen value. The distance of each pixel will compute first by the centtroid then it calculates the distance and velocity then calculates the pressure [6].

3.7 Optical flow HS Algorithm
This technique also perform the same phenomenon as lucas kannade algorithm but it removes the aperture problem as well as brightness contancy. It also accepts the video in AVI format. It calculates the intensity of each pixel then it further proceed to optical analysis block that where it calculate the vector and give results in colour formation with their intensity then it goes to thresold block that solves the velocity thresold and it analyze the shape of the object and apply binarization to the image.

```
d_dx = sqrt((U.^(2)-Uo) + (V.^(2)-Vo));
d_dt = 1/30;% d_dt = 1/framerate
d_v = d_dx/d_dt;
rho = 1.225 ; % Density of air = 1.225kg/m^3 @ SL
p_dyn = (0.5*rho*d_v.^2); % Dynamic pressure
%Cp = 1-(225./(d_v.^2)); % Coefficient of Pressure % Please check is

  subplot (2,1,1);
  plot(d_v);colorbar;
%   axis([0 30 0 ])
  xlabel('Velocity m/s')
  ylabel('intensity m/s')
  %CREATELEGEND(axes1)
% Create legend
legend1 = legend('vec1','vec2','vec3','vec4','vec5','vec6','vec7','vec8','vec9','vec10','vec11','vec12','vec1
set(legend1,...
    'Position',[0.0302586634822912 0.32131399889063 0.0614934107002367 0.561349677253355],...
    'LineWidth',1,...
    'Interpreter','none',...
    'FontAngle','italic',...
    'FontSize',8,...
```

Figure 19 LK and HS code calculating Velocity and pressure

```
legend1 = legend('vec1','vec2','vec3','vec4','vec5','vec6','vec7','vec8','vec9','vec10','vec11','vec12','vec1
set(legend1,...
    'Position',[0.0302586634822912 0.32131399889063 0.0614934107002367 0.561349677253355],...
    'LineWidth',1,...
    'Interpreter','none',...
    'FontAngle','italic',...
    'FontSize',8,...
    'EdgeColor',[0 1 0],...
    'Color',[0.8 0.8 0.8]);
title(legend1,'vectors data');
        subplot(2,1,2);
        %fun=d_v*d_dt;
        %z=int(d_v,d_dt);
        %z=int(fun,d_dt);
        % plot(d_dx);
        % scatter(p_dyn,x,y);
        %grid on; view([90,45]);
        %plot(d_dx);colorbar;
surf(p_dyn);colorbar;))
        xlabel('Dynamic Pressure (Pa)')
        ylabel('intensity')
```

Figure 20 HS and LK Code plotting Velocity and Pressure

3.8 How LK and HS Algorithm works

Table 01 shows the algorithms velocity analysis with the values of DIP flow LK and HS, DIP flow 1 and anemometer.

Algorithms	DIP FLOW LK	DIP FLOW HS	DIP FLOW 1	Anemometer
Velocity	13m/s	11m/s	12m/s	10.3m/s
Error in %	20 %	5 %	10 %	0 %

Table 01 Algorithm Velocity Analysis

3.9. Final Output of captured Video

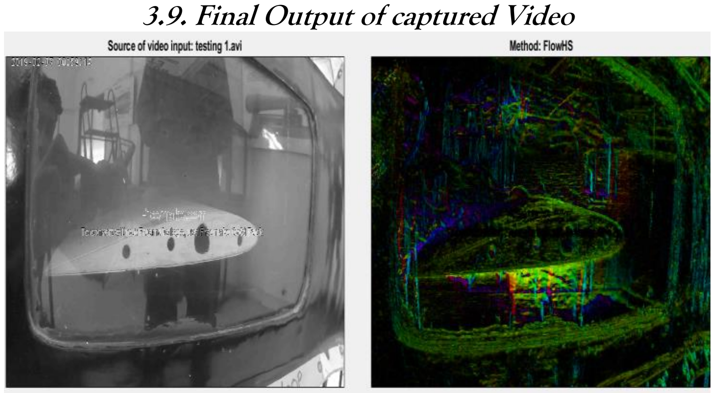

Figure 21 Output of Airfoil in HS colour Coded Form

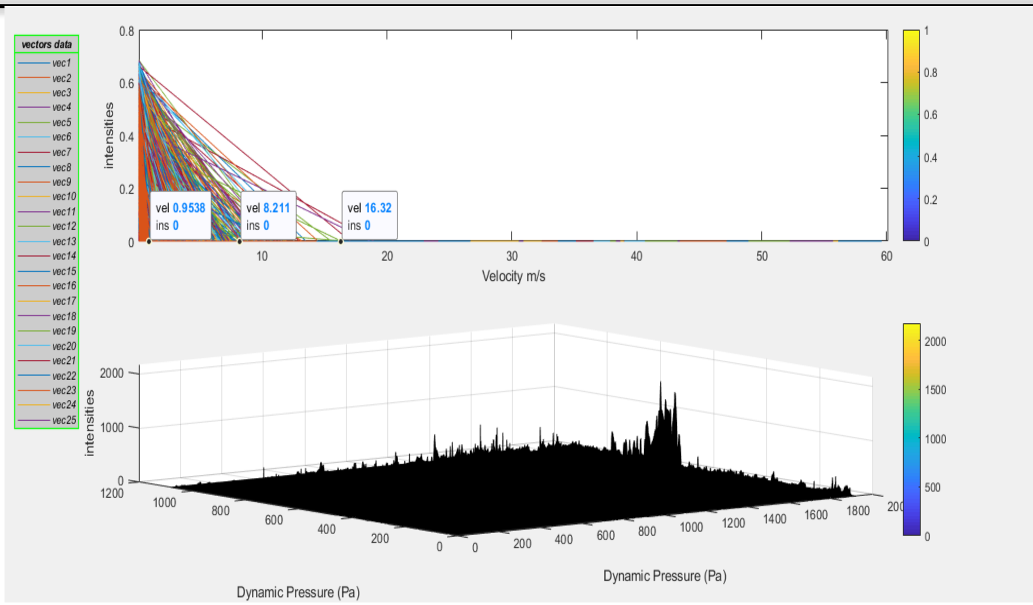

Figure 22 Velocity and pressure outputs of Airfoil at final stage captured video

4. Conclusion

The results of velocity and dynamic pressure obtained from the digital image processing using two algorithms LK and HS were verfied using digital anemometer. It was found that the results error percentage concerning the anemometer reading was arround 20%. However, the results of the flow velocities using HS were in high agreement with the anenometer readings, showing HS is better than other algorithms. Moreover, HS algorithm is computationally fast but its noise sensitivity is low, due to which it bright constancy is hard to bifurcate between low and high intensities.

5. Future Recommendations

In future, this project can be further optimized if there is a working on 3D videos. The main parameters are velocity and pressure, and from these two, other aerodynamic parameters can be obtained.

6. References

[1] S. Kato, S. Murakami, A. Mochida, S.-i. Akabayashi, and Y. Tominaga, "Velocity-pressure field of cross ventilation with open windows analyzed by wind tunnel and numerical simulation," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 44, no. 1-3, pp. 2575-2586, 1992.

[2] R. Howell, N. Qin, J. Edwards, and N. Durrani, "Wind tunnel and numerical study of a small vertical axis wind turbine," *Renewable energy*, vol. 35, no. 2, pp. 412-422, 2010.

[3] R. R. Schultz and R. L. Stevenson, "Extraction of high-resolution frames from video sequences," *IEEE Trans Image Process*, vol. 5, no. 6, pp. 996-1011, 1996.

[4] S. S. Beauchemin and J. L. Barron, "The computation of optical flow," *ACM computing surveys (CSUR)*, vol. 27, no. 3, pp. 433-466, 1995.

[5] D. Patel and S. Upadhyay, "Optical flow measurement using lucas kanade method," *International Journal of Computer Applications*, vol. 61, no. 10, pp. 6-10, 2013.

[6] D. Shukla and E. Patel, "Speed determination of moving vehicles using Lucas-Kanade Algorithm," *International Journal of Computer Applications Technology and Research*, vol. 2, no. 1, pp. 32-36, 2013.

[7] X. Gong and S. Bansmer, "Horn–Schunck optical flow applied to deformation measurement of a birdlike airfoil," *Chinese Journal of Aeronautics,* vol. 28, no. 5, pp. 1305-1315, 2015.

[8] D. Anfossi, D. Oettl, G. Degrazia, and A. Goulart, "An Analysis Of Sonic Anemometer Observations In Low Wind Speed Conditions," *Boundary-Layer Meteorology,* vol. 114, no. 1, pp. 179-203, 2005.

[9] Y. Huang, "Measuring flow in digital video containing smoke and gas," ed, 2011.

[i] https://www.semanticscholar.org/paper/Optical-Flow-Based-Moving-Object-Detection-and-for-Aslani-Mahdavi-Nasab/a3248c45cdc41417ed2ea236829a99c9783bb3ca

[ii] https://www.engineeringtoolbox.com/dynamic-pressure-d_1037.html