# A.N.V.I.S.: Automotive Night Vision Imaging System

Emmalynne Clarkston, Julian Ortiz, Carole Pearson and Rudy Valentin

Dept. of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, 32816-2450

***Abstract* — ANVIS is a prototype designed to increase drivers' visibility on the roads during low light conditions. By combining night vision, heads-up displays (HUDs), and voice activation a safer environment for drivers, passengers, and pedestrians can be created. The automotive night vision imaging system (ANVIS) was designed to be a multi-HUD system installed in the dash area of a vehicle that provides the driver with a 360-degree view of the surroundings displayed on the windshield. The voice activation allows the driver to have a hands-free means of switching between the views.**

## Introduction

Drivers often experience more difficulty driving at night due to the lower visibility. Because of the conditions, fatal accidents are three times more likely to occur at night [1][2]. ANVIS seeks to bridge the gap between the increasing technology of today's world. To do this, it uses infrared cameras to take in the car's surroundings. ANVIS utilized HUD technology to overlay the windshield with this infrared imagery. This document has been written to describe the development process of this design and the technology involved to create the system. First, the components of the design are discussed and the parts chosen are noted. Following this, the functionality of each part is discussed in more detail. Administrative content, including copyright permissions and sources used, are at the end of the paper after the final conclusions, if further study is desired.

## Design Overview

The function of this project is to be able to create an image overlay on the road and exterior environment ahead during low visibility conditions. The cameras are used to capture images of sides of the car. The primary HUD uses a near infrared camera at 850 nm, which is outside the visible light spectrum. The images are taken from the camera and displayed on a driver's windshield. This will allow the driver to

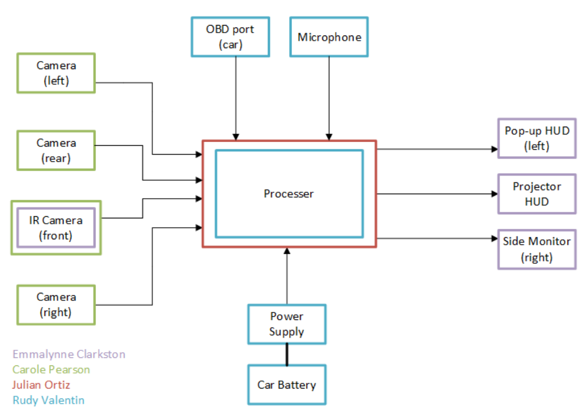


Figure 1: Block Diagram and Member task assignment

become aware of unseen obstacles and obstructions and allow the driver to be able to react in order to avoid hazards and drive more safely with the newly obtained visual information.

Our project idea combines optical, electrical, and systems engineering to create a multi-functional heads up display on a car's windshield. The design uses four cameras: 3 standard color cameras and one IR camera. The color cameras are on both sides and rear of the car. The IR camera is on the front of the car. To minimize the cost of the cameras while maximizing their efficiency, the cameras at the front of the car will be of higher quality than the other three sensors. The IR camera, as compared to the human eye, is able to display a much more accurate depiction of the area surrounding the car in low visibility conditions. The images from the IR camera will be layered through a series of image processes to enhance the view of the road. A processor will receive the data from the camera, apply the image enhancement algorithms, and then send it to the HUD located on the windshield for real-time use.

The HUD design will have two HUDs, one primary and one secondary. The primary HUD will use the processed IR image to display an outline of the road ahead on the windshield. This will be done using an optical configuration composed of lenses and mirrors with the image source being a projector. The HUD will be placed on the dashboard in front of the steering wheel out of the way of the driver's critical line of sight. There will be two main lenses in the configuration. Both are used to magnify the image and decrease the focal range while keeping the spherical aberrations to a minimum. The viewing screen will allow a reflected image to be seen on the windshield as close to the critical angle as possible. The secondary HUD will display the side views, rearview and diagnostics of the car. These will use LCD screens placed flat on the dash. The images will be directly reflected off of the windshield.

For the integration of voice control, a microphone will be used to gather audio data and use voice recognition APIs to process the user's voice commands to provide the desired output. The microphone will be a non-expensive, basic microphone that can simply gather audio with sufficient quality such that we can decipher what the user is saying. Then speech to text API is used to to convert the user's speech into text that the computer can read, and a program will be used to look for specific keywords such as 'left on' or 'right off' so it can execute the correct action. Speech API will also suppress the background noise and can help separate what the user is trying to say versus any random speech that is collected. This will help improve the functionality, accuracy, and success rate of the voice control feature for people that frequently drive with big, noisy families.

In a nutshell this project will contain both electrical components like most projects tend to have as well as software components. This will give a greater understatement to what being an engineer really is. The ability to have the exposure to both software and electrical design is a great thing since in the real world it's very important to have. This project is a very important tool in our lives that will hopefully lead to a much better understanding of the real work environment which all of the developers will need if they want to succeed in real life.

## Components

From figure 1 we can see that every single member of the group will play an important and equal role in the development of ANVIS. From the software to the hardware development each of the four group members will have the ability to work on both sides. Each member is assigned multiple tasks, some overlapping with other members tasks. This will give all the members in the group a look into the development of real world group engineering projects. This was one of the main goals when distributing the roles. Our mentor mentioned that the more experience that each of the members receives, the more each member will benefit from the project and be more ready for industry beyond academic level projects.

*1. Microcontroller*

The microcontroller used in this design is the Jeston Nano[3]. It has an HDMI connection capability for video output. The Jetson Nano uses a quad-core CPU at 1.43 GHz frequency which is necessary for the image processing demands of the project while keeping a real-time image output. These two attributes made the Jetson Nano the optimal selection when compared to higher-end and lower-end processors.

*2. PCB*

ANVIS will have multiple options when the input power is designed. The two requirements for battery power are that it can (1) power ANVIS for at least twelve hours continually and that (2) the total power consumption of the system will be twenty-five watts or less.

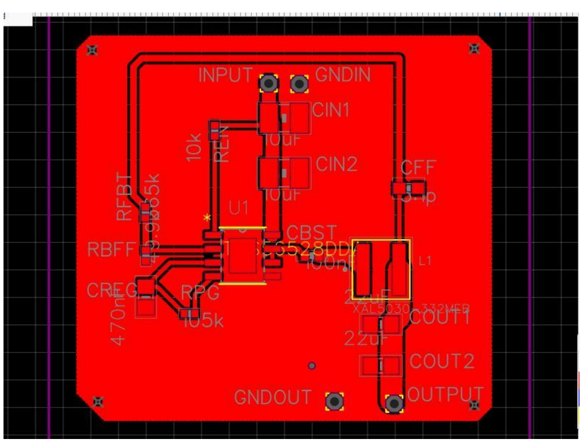


Figure 2: PCB Design

Designing a DC-DC converter as part of the project and using it as the required PCB design is the preferred option. In this application, designing a PCB with a DC-DC converter will allow ANVIS to be seamlessly integrated into the car's system, and does not add excess size and weight to the design. The schematic design uses the TPS56528 chip [4] and was designed using the WEBENCH Power Designer [5]. The PCB design was created in Eagle.

*3. Microphone*

To implement voice recognition into the system an embedded system will be used. This is because an embedded system requires minimal human interaction and ANVIS is only meant to be able to handle a small amount of specific functions. The only human interaction the voice user interface needs is the input audio of the driver when they say a command. A peripheral USB microphone [6] is used to receive the input and then the code will run to produce the output needed.

Since the microphone is connected through a USB port on the processor board, the USB communication protocol will be used. The microphone connection is USB 2.0, but this will still be fast enough to meet latency requirements for the time between audio input and system output. The microphone will record audio at 16 kHz frequency since this is the ideal frequency that the voice model being used can process audio.

*4. On-Board Diagnostics (OBD)*

The OBD Port on the car gives a user access to the diagnostic such as speed, rpm, error codes, etc.. The OBD information it detected through serial devices. This device connects to the processor through a USB connection and then the correct protocols are detected using commands in pythons OBD package. The package comes with predefined explanations for most DTC codes that can be returned by the check engine function, however it might not contain all special codes from the manufacturer.

In this case it will just return the code with no explanation. For these a manually inputted explanation from the manufacturer's documentation is available.

*5. Cameras*

When choosing what types of camera to buy, there are certain characteristics that must be evaluated. ANVIS will have four camera/sensors in total: a rear/backup camera, two side cameras, and an IR camera in the front. Cameras were chosen based on the field of view and the distance captured. Our IR camera [7] has a field of view of 60 degrees and a 10 meter working distance. The two side cameras [8] have a field of view of 35 degrees and a 5 meter working distance. The backup camera [9] has a field of view of 120 degrees and a 5 meter working distance.

*6. Projector*

The image source for the primary HUD is the Kodak Luma 75 [10]. This is a DLP-based projector which allows it to be a relatively small projector. With its compact size, it can easily fit on the roof of the car near the visor of the car without obstructing the driver's line of sight. It is also very light, 12oz, which made it easily mountable. Since it is a DLP, it does not have as high of a contrast ratio, but a ratio of 1300:1 is acceptable for our application. The focal range for this projector is at a relatively close distance starting at 10 inches.

*7. LCDs*

The LCD screens were chosen based on the ease of compatibility with the Jetson Nano MCU [11].They were chosen because they came already premade to work with the Jetson Nano and would be the easiest to integrate into ANVIS. The resolution of the image on the LCD is 1024x600. They are powered directly from the Jetson Nano by and they connect to the Jetson Nano by way of an HDMI port.

*8. Lenses*

The two lenses used in the optical system are both the MSCN020HAR.11 negative meniscus lenses. The MSCN020HAR.11 has an aperture size of 2 inches. Its effective focal length is -500 mm. The lens has an average refractive index of 1.459 in the visible light spectrum. Also, the concave-convex design of the lens design of the lens reduces the amount of spherical aberrations that would be seen in the final image. [12] [13]

*9. Image Processing*

When deciding what type of image processing needed for the project, we started by figuring out what our end goal was that we wanted to achieve, in the beginning we had a faint picture of what we wanted. We needed to take an IR image and process it in a way that would make the dark pixels brighter while at the same time keeping the light pixels the same. Pixels are the building blocks of an image, they are the smallest controllable element of a picture represented on a screen. The more pixels you have the more your image will look like to the original. Pixels have a color spectrum which ranges from 0 all the way to 255 due to the fact that each color that is used as a building block of a pixel has 256 shades. The three categories of colors that build up a pixel are green, blue, and red. These colors have their own bands which have their wavelength characteristics. Since we are using IR imaging we use band 5 which has a wavelength that ranges from .85-.88 micrometers. This wavelength determines the intensity of light that we are using. The longer the wavelength the weaker the light will be and the shorter the wavelength the stronger the light will be.

When manipulating pictures there are really only two things that we can change, one being brightness, and the other being contrast. To be able to know what to change we have to know what each one is. Brightness is the attribute of visual representation in which a source appears to be radiating or reflecting light, meaning that it is the amount of light that an image reflects. Contrast is the difference in color that makes an object distinguishable. We began by creating an algorithm that changes the brightness and contrast of our IR image, we chose to write in python since it was one of the simplest languages we could use for image processing. We began by finding out what type of libraries existed that had pre existing properties for what we desired. We found out that OpenCV had a pre-existing function that manipulated brightness and contrast with the changing of two variables, we will discuss these two variables soon. The way that OpenCV manipulated the image was by using point operators which are what are needed for pixel transformation and well as neighborhood operations which is what transforms certain areas in the image. This all gets put together into an equation, $g(x) = \boldsymbol{\alpha} f(x) + \boldsymbol{\beta}$. Alpha and beta are called the gain and bias parameters, these parameters control contrast and brightness respectively.

When manipulating a projection you tend to have distortion that occurs thanks to having a double image appearing. This is a big issue in our project since we need to be able to project onto a mirror without having the issue of double imaging. This can be solved by certain methods. You can either create a filter that does not allow for that distortion to occur or you can create an image processing software that takes care of it for you before you project it onto the screen. Since we determined the problem we began to try and solve this issue by first searching to see what has been done before to help fix this. We found out that the main issue that needed to be fixed was the shift that was created by the input when projecting the image onto the screen, this has an easy solution which is to reduce the thickness of the glass that the output image is being projected onto. We need to use the following equation to find out how thick the glass needs to be. with the following constraints. Z is the shift distance between the two images and m is the number of pixels within the distance that gets shifted. The matrix is the shift matrix, what this does is it calculates the secondary virtual image based on the primary virtual image. If we multiply the shift matrix with the image we shift certain pixels by m which would then allow us to get a much more focused image at the end. Keep in mind that X has to be greater than or equal to 0 if we do not take this seriously we end up trying to display negative luminances which no physical device can actually achieve.

## HUDs

The HUDs use the windshield to reflect an image so that it is visible to the driver. The design is made to be installed to the specification of the Honda 2006 CR-V. The Primary HUD goes through an optical setup with its source being the projector. The Secondary HUDs just use a pure reflection of their source the LCD screens. In order to get the optimal output image for all three setups, they each need to be characterized by the materials they reflect off of and pass through in their optical path. The final optical system will be installed to the scaled representation seen in figure 3.

Figure 3: HUD Design

1. *Primary HUD*

The design for the main HUD starts with the projector. It is mounted on the ceiling of the car, perpendicular to the windshield. The image is first reflected off of a 2 inch square mirror. The mirror allows for 5 inches to be added to the optical path length. This increases the potential projection size of the final image. The mirror inverts the image the first time. Next, the image passes through two MSCN020HAR.11 lenses. These lenses increase the output image size while limiting the amount of spherical aberrations. The lenses also add a small amount of blurring to the image, which is a method used to reduce the appearance of the ghost image on the windshield. After the lenses, is the propagation distance of 12 inches to the viewing screen. The viewing screen creates a flat plane for the image to focus on. After the image is formed on the viewing screen, the driver is able to perceive it as a reflection off of the windshield. The image is inverted again at this point, reverting it back to its original orientation. The height, width, and angle of the image will be congruent to that of the viewing screen. The driver's eyes will form an image of the reflection beyond the windshield relative to the distance between the windshield and viewing screen, thus creating the main HUD. The adjustment of the projector, lenses, projections distance, and viewing screen plane will allow for the projected image to align with the driver's initial view. This will create the HUD’s night vision effect on the road.

The original image taken from the IR camera displayed on the windshield does not perfectly overlay the road ahead initially. The image needed to be resized to fit the display size. The perspective of the image also needed to be adjusted to fine-tune the image overlay of the road.

*2. Secondary HUD*

The secondary HUD’s design is much simpler than that of the primary HUD’s design. These start with the LCD source. These are placed to the right and left of the primary HUD’s viewing screen. The driver is able to perceive a reflection of the LCDs as a reflection off of the windshield. The height, width, and angle of the image will be congruent to that of the LCDs. The driver's eyes will form an image of the reflection beyond the windshield relative to the distance between the windshield and LCDs, thus creating the secondary HUDs. Since the output of these images comes from the side, rear and diagnostics display with none being relative to the forward facing view, they do not need to have the fine adjustments of the primary HUD. These HUDs will sit on a plane relatively parallel to the ground plane. This angle reduces the appearance of the ghost image on the windshield.

## Cameras

ANVIS will be using four cameras in total, an IR sensor that is faced looking forwards, a regular camera on the left hand side that will be viewing the left blind spot, another regular camera on the right that will be viewing the right blind spot, and a rear view camera that will be placed in the back of the vehicle. These cameras are each connected to the MCU. The MCU will process the image data so that it can be seen on the HUD displays. The IR camera will be used at night and will be displayed using the projector on the windshield. The side cameras and back cameras will be options to be displayed on the LCD screens when prompted by the user.

## Power Source

The main power source is the car battery, which operates from 10V when the car is off to 16V when the car is on. The car battery will feed into our PCB through the fuse box, which will then convert to the necessary 5V to power our MCU. The Jetson Nano in turn powers everything else.

## Voice activation

Voice activation will be used to turn on or “wake up” ANVIS. The way the voice command code will work is once the trigger word is detected, the software will then start recording the audio from that point. Then python commands will be used to detect certain keywords in the upcoming sentence that will tell the system which settings to change. Once the code determines the settings that are to be changed, it sifts through the rest of the audio to detect the desired value to set that specific setting to. This could be a numerical value or it could be telling the system to turn a certain feature on or off.

## Testing

The following section contains the procedures taken to test the project as a whole system. It will contain exact procedures that were conducted for high quality outcomes. The equipment used in the test is ANVIS, a desktop PC with internet connection, plexiglass, and a dark environment setting.

To test ANVIS as a whole unit a dark environment will be needed to be able to test correctly. We need to create an environment that resembles the inside of a vehicle in the most accurate way possible. Testing ANVIS within a car is much more difficult since it creates a more unstable testing environment. Because of this we will set up a functioning test bench setup that will be structured with the dimensions of our vehicle to simulate the working environment.

Test 1: We begin by testing the projector quality with the image that the IR sensor is sending to the microcontroller. We use a piece of plexiglass to resemble the windshield that the image will be projecting onto, this will give us enough data to be able to work out specific errors we might have with the angle needed to get a perfect image being projected. With the dark environment it gives us a much better environment to test, we want to make sure that we have the correct amount of lighting being displayed from the projector onto the piece of plexiglass so that in the real world the user can fully see the image being projected in front of them.

Test 2: We begin by testing all the algorithms as a whole unit. We combined both the image processing algorithms with the voice recognition algorithms to create a single algorithm that will control what will be projected. We test each section of the algorithm, we start with making sure the detection algorithm is working by using the voice recognition algorithm mixed in with the detection algorithm to detect when a specific value for contrast or brightness is said. This will mix in with the image processing algorithm to change the displayed image accordingly to what was detected using the detection algorithm.

Test 3: We begin by testing all the main components' reactions when mixed with all the algorithms. We use the voice recognition algorithm to turn the right and left LCDS either on or off as well as the rear view camera on or off. We use the detection algorithm discussed in test 3 to detect when each of the cameras are needed to be turned on depending what the user wants.

Test 4: We begin by testing the OBD, we use both voice recognition algorithms as well as the detection algorithms to test if the OBD is working the way it needs to work. We make sure that the information the user wants to be displayed is correctly displayed and can easily be removed if the user chooses to do so.

Test 5: The final test we conduct is making sure the whole system works as a whole. We try to mix all the tests into one major test to make sure everything works simultaneously. We first make sure that the system is getting the correct power, if this isn't the case nothing would work as expected. We then follow this by making sure that all the viewing angles are correct, we do not want a user to be miscommunicated by what they're seeing. We then make sure all the algorithms are working as a whole unit, we test each one individually then all of them at the same time. We end by making sure that the system is fully cased in and secured.

## Conclusion

ANVIS began with the idea of an intelligent device capable of displaying important information onto the windshield of a vehicle. These ideas stemmed from increased probability of accidents during low visibility conditions This created a want in development of a device that could prevent these issues from occurring. From there the development of ANVIS began and can be developed into a product that can help lives worldwide.

ANVIS possesses all of the key technologies needed for protecting drivers in low visibility conditions as well as in daily driving. Through the documentation that has been provided that encapsulated all of the necessary information for other engineers to replicate the design. References have been made throughout the document to patricide what the industries do. ANVIS serves as a demonstration of many concepts that are key in making driving a much safer environment.

## Biography

Emmalynne Clarkston is an optics engineering student. She designed the optical system, image processing code for the primary HUD, and mounts and casings for all the components.

Julian Ortiz is an electrical and computer engineering student. He designed the main control code for the system.

Carole Pearson is an electrical engineering student. She designed the PCB as well as the camera system.

Rudy Valentin is an electrical engineering student. He designed the voice activation code and the OBD communication.

## References

[1] Road traffic casualties: understanding the night-time death toll
[2] https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/810637
[3] https://www.amazon.com/NVIDIA-Jetson-Nano-Developer-945-13450-0000-100/dp/B084DSDDLT
[4] https://www.ti.com/lit/ds/symlink/tps56528.pdf?ts=1617994144761
[5] https://webench.ti.com/power-designer/switching-regulator/export/8?noparams=0
[6] https://www.amazon.com/Microphone-MAONO-Omnidirectional-Microphone-Recording-Broadcasting/dp/B074BLM973
[7] https://store.spinelelectronics.com/UC20MPD_ND
[8] https://www.amazon.com/Creative-Labs-73VF064000000-LiveCam-SOCIALIZE/product-reviews/B005WH9UAM
[9] https://www.amazon.com/Microphone-Unzano-Computer-Streaming-Conference/dp/B08GPZK2ZQ
[10] https://www.amazon.com/gp/product/B078NCG82N/ref=vp_c_AU9QZC5SYU88D?ie=UTF8&m=A2LM6ZPY06LT1N
[11] https://www.waveshare.com/jetson-nano-developer-kit-package-c.htm?sku=17704
[12] https://www.newport.com/p/MSCN020HAR.11
[13] https://www.newport.com/p/KBC070AR.33