SMART ADAPTIVE VEHICLE LIGHTING SYSTEM

Ch Prithviraj[1], K. Archana Bhange[2]
[1]PG Student, [2]Sr. Assistant Professor
[1,2]Dept. of ECE, VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad, India

Abstract
These days the number of accidents is increasing tremendously, especially at night. The main reasons for such accidents are improper lighting conditions and glare of the vehicle headlights. Hence this project "Smart Adaptive Vehicle Lighting System" (SAVLS) gives solution to this problem. In the proposed system two features are implemented. They are (i) Adaptive Control of Brightness of Headlights (ACBH) (ii) Remote Control of Vehicle Headlights (RCVH).

Brightness of headlights of a vehicle is controlled adaptively based on input from camera that is headlights of the oncoming vehicle as well as headlights movement is controlled based on the steering wheel sensor movement, by using the Raspberry pi3 model B board and processing is done in Open Camera Vision (CV).

Headlights are controlled remotely by making use of Raspberry pi as an Apache Web Server and accessed by either smart phone or personal computer (PC) through a web browser with internet connection using its IP Address.

Keywords: Smart Adaptive vehicle lighting system (SAVLS), Raspberry pi3 model B, Open Camera Vision (CV), Apache Web Server, Smart phone, IP Address.

I. INTRODUCTION

As advancement in vehicular technology over the years the number of manufacturing of vehicles also increases vastly because of its usage in our daily life. According to the survey it is evident that there is going to be 67% of increase in number of accidents by 2020, mostly at night time. In comparison, the percentage of major accidents occurring at night time is 1.5 times more than that of a daytime and also it shows that 60% of the accidents take place on the curve roads than other roads [5] due to improper lighting conditions. Reasons for such accidents are due to improper visibility of pedestrians at the curved paths, poor lighting conditions and glare of oncoming vehicle headlights, blind spots at the curved paths and also steep turns in remote areas like hills and mountains which may lead to the accidents.

When oncoming vehicle headlights fall on our eyesight, the driver may face blindness for short period of time, which results into a situation called Dazzling of headlights, which can cause a serious threat and results in loss of many lives. So, appropriate illumination of road to the drivers is a major task. Hence headlights play a vital role in preventing such accidents.

Due to static headlamp systems blind spots will occur, which need to be avoided for road safety. Therefore, a new dynamic technology called "Smart adaptive vehicle lighting system" (SAVLS) is designed to address the solution to this major problem.

II. LITERATURE RELATED

Over the few decades many Adaptive front lighting systems were designed.

In the work proposed by G. Dong, W. Hongpei, G. Song, and W. Jing [7] is based on steering wheel & speed sensor as well, depending upon the road conditions headlights of vehicles are adjusted. According to the work proposed by Yali Guo, Qimnu Wu and Honglei Wang [6], it is evident that dynamic adjustments of headlights are done by using motion models and also with necessary mathematical equations. The turning angles of headlights in both the directions are calculated in order to adjust the range and irradiation direction of headlights. The work of Shreyas S, Kirthanaa Raghuraman, Padmavathy AP, S Arun Prasad, G. Devaradjane [3] shows that headlamps which are mounted on stepper motors gets automatically switched to ON/OFF state based on the light intensity measured by the photodiode and also through rotation of steering wheel and is based on the indicator switch, which acts as input to the system. From the work of Jyothiraman De [4] LED's which are connected to relay circuit are controlled and are based on appropriate actions carried out by comparator circuit and the rotation of steering wheel and also beam angle is adjusted dynamically. The work of Fengqun Guo, Hui Xiao, Shouzhi Tang [5] is about photometric models based on Charge-Coupled Device (CCD) image sensor technology which gives curvature radius information by making use of Electronic Circuit Unit (ECU) and simulation is done in MATLAB. The work of Hui Rong, Jinfeng Gong, Wulin Wang [8], kinematics model system for both vertical plant and horizontal plant were designed and control strategy for AVLS is carried out and also fault management of system is implemented. According to Snehal G. Magar [1] work is based on image processing technology where oncoming vehicle is detected by using ultrasonic sensor and also camera acts as an image sensor to detect the curvature angle of white line of road.

III. PROPOSED SYSTEM DESIGN

In this project SAVLS system design is done based on two requirements
(1) Hardware Requirements
(2) Software Requirements

(1) Hardware Requirements

Hardware requirements of SAVLS are developed by using:

a) Raspberry pi3 model B:
Raspberry pi3 acts as a micro-computer for main processing and measures the light intensity from the input given by the camera, in order to control the motion of the servomotor and also brightness of headlights.

b) Servomotor (GS-3630BB):
Basically Servomotor works on the Pulse Width Modulation (PWM) signal, its angle of rotation is controlled by duration of applied pulse to its signal output pin through a variable resistor i.e., rotational potentiometer. When potentiometer is rotated PWM signal is generated by the controller and then servomotor movement takes place according to the given signal. So that headlights on servomotor get rotated.

c) Web Camera (QH495LM):
Camera used here is capable of night vision with 6 bright lights, which is used to monitor the road and captures the headlights of an oncoming vehicle.

d) LED Bulb:
Light Emitting Diodes bulb acts as headlights of vehicles. LED's are preferred because of its low power consumption, high luminous intensity, more reliability and long life. LED's are mounted on the top of Servomotors and gets rotated simultaneously with the Servomotor rotation. The brightness of LED's is controlled by the input from the Raspberry pi given by the camera.

e) A/D Converter (MCP-3008):
MCP 3008 is a 2.7V 8-channel 10 bit A/D converter with SPI serial interface. Here A/D Converter is used to convert the analog value into digital value of light intensity in order to give input to Raspberry pi.

f) Rotational Potentiometer:
10K Rotational potentiometer acts as steering wheel sensor. Usually potentiometers are variable resistors and they function to alter their resistance via a knob or dial. Here potentiometer is used for adjusting the movement of servomotors.

g) Power Supply:
Power supply is provided to the system in two ways (i) A 12V Lithium-Ion rechargeable DC battery is connected externally to the system (ii) A +5.1V micro USB cable based charger is used and connected externally to the power supply port of Raspberry pi board.

(2) Software Requirements

Software requirements of SAVLS are developed by using:

a) Raspbian Jessie with Pixel:
Raspbian Jessie is a free open source operating system based on Debian distribution. It is booted

in the raspberry pi to make interaction between the user and the whole system.

b) Python Programming Language:

Python is a flexible, high-level, interactive, interpreted, highly readable, object-oriented scripting language and also runs on different standards editors like leafpad, nano or vim. Programming of Raspberry pi is done using python programming language in order to control the brightness of headlights.

c) HTML:

HTML stands for Hyper Text Markup Language. Using HTML, a web page is defined and also it tells web browser how to display content in the web page.

d) CSS:

CSS stands for Cascading Style Sheets. CSS is a technology used to design websites and plays a key role in presentation i.e., colors, fonts, layouts, and overall look of the website.

e) Apache Web Server:

Apache Web Server is free and Open-Source software. It is designed to run web applications. Apache Web Server takes the request from the user and processes it and gives response to the user.

f) PHP:

PHP stands for Hyper Text Pre-Processor. It is a dynamic web technology runs at the server side i.e., Apache web server. PHP scripts are used for accessing the whole system where it takes the user data and performs validation at the server side and gives the result back to the user.

IV. SYSTEM OVERVIEW AND IMPLEMENTATION

The overall architectures of SAVLS for both the features

(i) Adaptive Control of Brightness of Headlights (ACBH)

(ii) Remote Control of Vehicle Headlights (RCVH) is implemented below.

(i) Overview of Adaptive Control of Brightness of Headlights (ACBH):

A) Description of block diagram

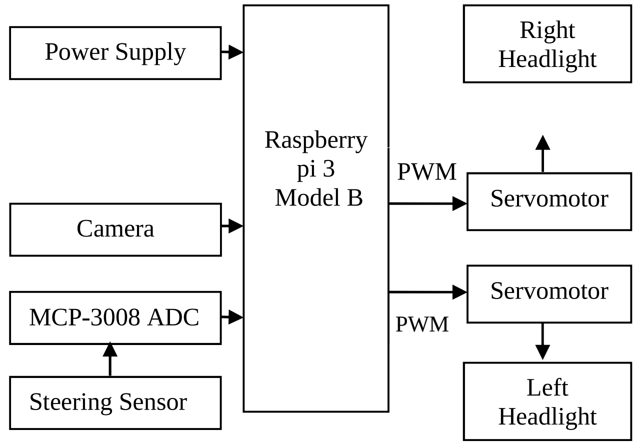

Figure 1: Block diagram of ACBH

The headlights are placed at either ends of the vehicle and camera is placed in between them that is at the centre. The two headlights and camera are mounted on three different Servomotors individually and are controlled by the Raspberry pi. Here the steering wheel of a vehicle is attached to the steering wheel sensor that is 10k rotational potentiometer and sends the PWM signal which is generated by the Raspberry pi board to servomotors for controlling the movements of the headlights and also camera is used to capture the headlights of the oncoming vehicle and also captured image is fed as control input to the Raspberry pi board through SPI communication where processor process the captured image using open CV and also MCP-3008 ADC converter converts the light intensity of captured image into digital value and sends the data to the Raspberry pi board and then controls the brightness of headlights accordingly to reduce the glare and illuminate the path properly.

B) Flow Chart

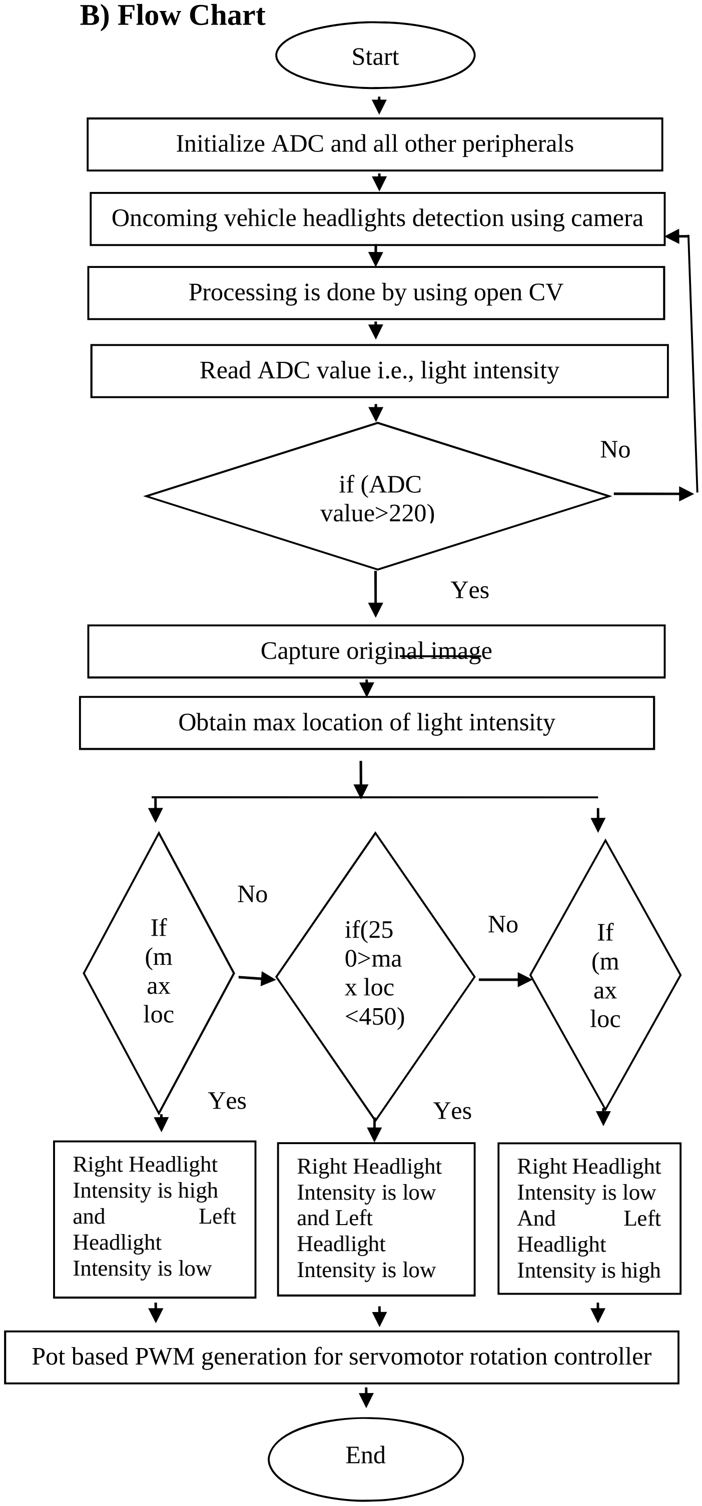

Figure 2: Flow chart of ACBH

As shown in the Figure 2, initially ADC and all the other peripherals of the system are initialized, monitoring of road is done by using camera. Oncoming vehicle headlights are detected by using camera and also processing is done using open CV in order to calculate light intensity. Light intensity value is read by the ADC. If the obtained ADC value is greater than 220, camera captures the original image and performs appropriate image processing operations to obtain the maximum location of light intensity in the captured image. If the maximum location of light intensity is less than 450 then duty cycle of right headlight intensity is high and left headlight intensity is low. If it is between 450 to 750 then right headlight intensity is low and left headlight intensity is also low. If it is greater than 750 then right headlight intensity is low and Left Headlight intensity is high. And also PWM signal is generated by Raspberry pi based on potentiometer movement which drives the servomotor accordingly.

C) Software

In the software part, initially we need to boot Raspbian Jessie with pixel operating system which is linux based into SD card and also open CV is installed for image processing where various operations likes RGB to Gray conversion, Gaussian blur algorithm is performed to obtain the maximum position of light intensity value from the captured image by the camera and programming is done using python language. In RGB to Gray conversion, a color image is converted into gray scale image and in Gaussian blur algorithm intensity values of lighter areas of an image are connected to binary (1's) and intensity values of darker areas of an image are connected to binary (0's) in order to obtain the maximum position of light intensity value and makes a blue circle in the image at maximum location with radius 41 and thickness 2. Based on the light intensity value duty cycle of headlights are adjusted accordingly to control the brightness of headlights.

(ii) Overview of Remote Control of Vehicle Headlights (RCVH):

A) Description of block diagram

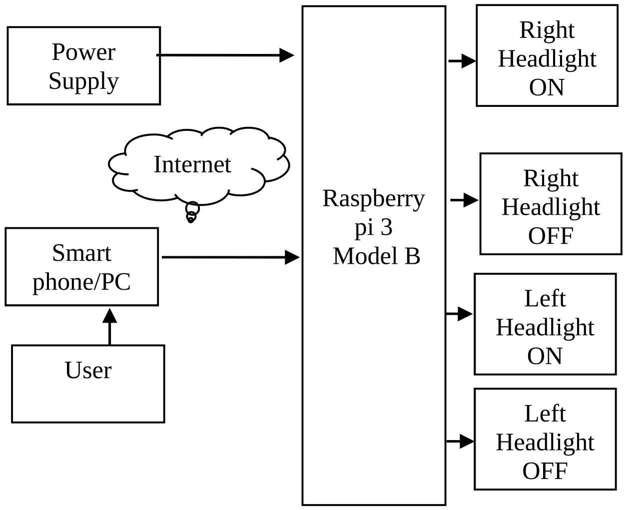

Figure 3: Block diagram of RCVH

Remote control of headlights is done by the user from a remote location by making raspberry pi as a Apache2 Web Server and the whole system is accessed through a smart phone (or) PC from a web browser using its IP address connected to the internet via Wi-Fi, 3G/4G/5G network or Ethernet etc., and thus according to the input given by the user through the web page, raspberry pi will change the headlights of a vehicle to ON/OFF state.

B) Flow Chart

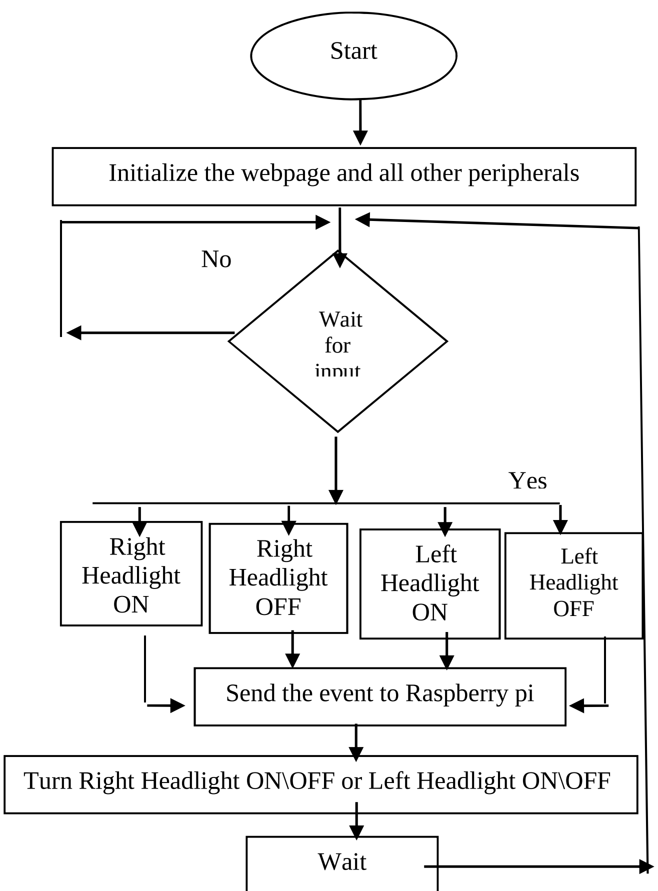

Figure 4: Flow chart of RCVH

As shown in the Figure 4, initially the virtual web page is initialized and the system is powered on. And the hardware waits for the input event to occur. If NO i.e., event from web interface does not occur it will be in this loop waiting for the input event to occur. If YES i.e. event from web interface occurs, the event will indicate the raspberry pi. The raspberry pi will act accordingly to the input event and gives the desired output.

C) Software

Remote access of vehicle headlights is implemented by installing apache2 web server in Raspberry pi board. Initially Graphic user interface (GUI) is designed by using programming language like HTML and CSS where accessing of headlights is done with the help of web browser of smart phone (or) PC by running PHP script on it.

V. EXPERIMENTAL RESULTS

The output of SAVLS includes two sections.

(i) Adaptive control of brightness of headlights (ACBH):

Table 1: Shows Light Intensity values of ACBH

Light Intensity Location Range	Right Headlight Intensity	Left Headlight Intensity
0-250	High	Low
250-450	Low	Low
>450	Low	High

Based on maximum location of light intensity, brightness of the headlights of vehicle is adjusted by the raspberry pi as given in table 1.

Figure 5: Output of ACBH

As shown in the Figure 5, if the maximum location of oncoming vehicle light intensity is 192, it is evident that a blue circle is made on that image and appropriate action is taken by the controller and makes Right headlight intensity to high and Left headlight intensity to low.

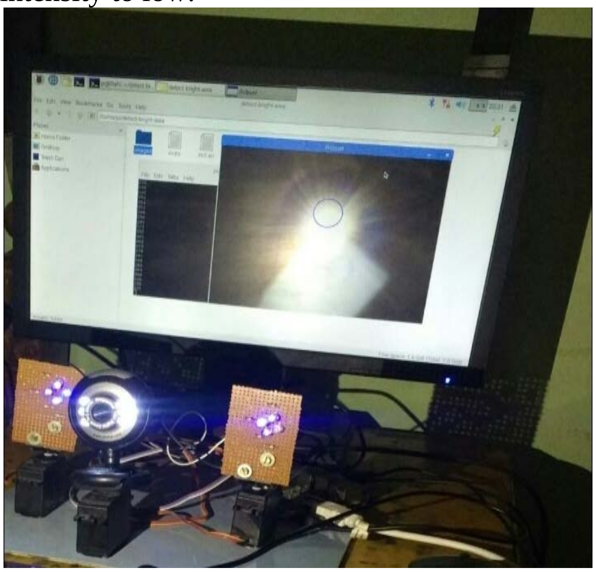

Figure 6: Output of ACBH

As shown in the Figure 6, if the maximum location of oncoming vehicle light intensity is 425, it is evident that a blue circle is made on that image and appropriate action is taken by the controller and makes Right headlight intensity to low and Left headlight intensity to low.

Figure 7: Output of ACBH

As shown in the Figure 7, if the maximum location of oncoming vehicle light intensity is 569, it is evident that a blue circle is made on that image and appropriate action is taken by the controller and makes Right headlight intensity to low and Left headlight intensity to high.

(ii) Remote control of vehicle headlights (RCVH):

Headlights of a vehicle are controlled by accessing the system using its IP address from a web browser through GUI interface as shown in the following Figures.

Figure 8: Output of RCVH

As shown in the Figure 8, when input is given from the user through web interface by clicking on the Right Light ON, apache server receives the request from the browser and at the background it processes the PHP code. According to the PHP script, raspberry pi will turn on the right headlight of a vehicle.

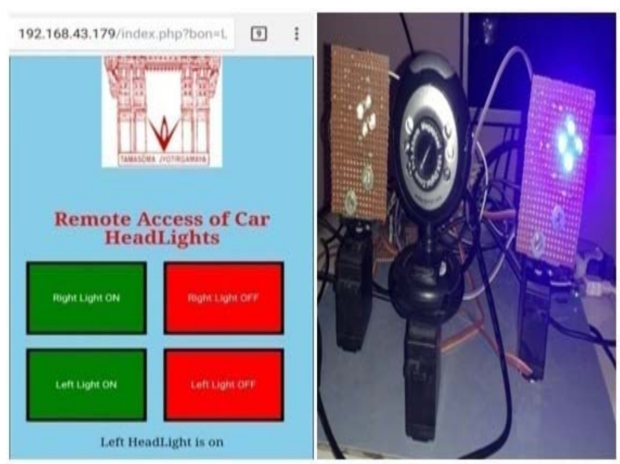

Figure 9: Output of RCVH

As shown in the Figure 9, when input is given from the user through web interface by clicking on the Left Light ON, apache server receives the request from the browser and at the background it processes the PHP code. According to the PHP script, raspberry pi will turn on the left headlight of a vehicle.

VI. CONCLUSIONS AND FUTURE SCOPE

In this paper, primarily adaptive control of brightness of headlights is done by capturing the image of oncoming vehicle headlights and then calculating light intensity. Based on light intensity brightness of headlight is adjusted in order to avoid glaring and illuminate the path properly. Secondly Remote access of headlights of a vehicle is successfully controlled through smart phone connected to internet where it utilizes PHP based web services. This feature will help the user to find the vehicle whenever there is no proper lighting condition in the parking lane/cellar. The proposed design has been tested and implemented in real time. This system when integrated with existing vehicle features can be drawn into a commercial product.

ACKNOWLEDGMENT

We would like to express our gratitude and sincere thanks to Dr. Y. PADMA SAI, the Head of Department of Electronics and Communication Engineering and to the VNR VJIET Management for their support throughout the project.

REFERENCES

[1] Snehal G. Magar, "Adaptive Front Light Systems of Vehicle for Road Safety" 2015 international Conference on Computing Communication Control and Automation, PP: 551-554 (IEEE 2015).

[2] Jiae Youn, Mewng Di Yin, Jeonghun Cho, "Steering Wheel-based Adaptive Headlight Controller with Symmetric Angle Sensor Compensator for Functional Safety Requirement" 2015 IEEE 4th Global Conference on Consumer Electronics (GCCE), PP: 619-620 (IEEE 2015).

[3] Shreyas S, Kirthanaa Raghuraman, Padmavathy AP, S Arun Prasad, G.Devaradjane, "Adaptive Headlight System for Accident Prevention" 2014 International Conference on Recent Trends in Information Technology, PP: 1-6 (IEEE 2014).

[4] Jyotiraman De, "Universal Adaptive Headlight System" 2014 IEEE International Conference on Vehicular Electronics and Safety (ICVES), PP: 7-10 (IEEE 2014),

[5] Fengqun Guo, Hui Xiao, Shouzhi Tang, "Research of Modeling and Simulation on Adaptive Front-Lighting System for Corner Based on CCD" 2013 25th Chinese Control and Decision Conference (CCDC), PP: 3598-3602 (IEEE 2013).

[6] Yali Guo, Qinmu Wu, Honglei Wang, " Design And Implementation Of Intelligent Headlamps Control System Based On CAN Bus" 2012 International Conference on Systems and Informatics (ICSAI 2012), PP: 385-389 (IEEE 2012).

[7] Guo Dong, Gao Song, Wang Hongpei, Wang Jing, "Study on Adaptive Front-lighting System of Automobile Based on Microcontroller" 2011 International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE), PP: 1281-1284 (IEEE 2011).

[8] Hui Rong, Jinfeng Gong, Wulin Wang, "Kinematics Model and Control Strategy of Adaptive Front Lighting System" 2009 Second International Conference on Intelligent Computation Technology and Automation, PP: 70-74 (IEEE 2009).