Indoor Positioning with FDM Coded RGBLEDs and Smart Phones

Guangtao Xue
School of Electrical and
Computer Engineering
Shanghai Jiao Tong Univ.
Shanghai, China 200240
Email: xue-gt@cs.sjtu.edu.cn

Guang Yang
School of Electrical and
Computer Engineering
Shanghai Jiao Tong Univ.
Shanghai, China 200240
Email: glfpes@sjtu.edu.cn

***Abstract*—**With the rapid proliferation of camera-equipped smart devices (*e.g.* smart phones, pads, gearings), visible light method as a novel way to suffice indoor positioning at mega malls or airports is appearing to be a reliable one since it provides high precision and with hardly additional peripherals excepts existing indoor LED luminaries compared with existing indoor positioning systems exploiting radio-frequencies that may defective at precision or RFID and other hardware-based approaches which needs rich deployment costs.

To achieve this goal, existing methods exploits the frequency domain to convey distinct landmarks. However, this relies on conditioned controlling of several rolling shutter camera parameters such as exposure time and is strictly limited to the highest exposure frequency since a camera can only identify different blink frequencies with sufficient small intervals parting them apart.

We describe our solutions that to address challenges mentioned above by exploiting a FDM coding mechanism to indicate multiple landmarks. After we determine the landmark, we can find a coarse positioning result collected from a digital map. We can introduce Angle of Arrival positioning algorithm to get a precise location as the result. Our prototype implementation demonstrate that our solution can offer an obviously promotion in the number of location landmarks compared to existing VLC based indoor positioning system under similar circumstances.

I. Introduction

Indoor localization have increasingly significance in modern indoors scenarios since the building cover prevents the availability of GPS satellite positioning signals. We believe that for most mega malls or large airports, a precise and user-friendly positioning system would be mightly valuable since customers or passengers can be lost in a complicated indoors environment. Besides, a mall equipped with a positioning system can deliver guided recommendation of merchandise for customers who walks near it. Location information is also required in the modern wireless sensor network based remote health monitoring systems.

As navigation and recommendation applications must rely on an indoor positioning system, We can figure out that a well designed indoor positioning system must satisfy at least four characteristics: 1)enough precise; 2)user friendly, which means user can attain the positioning results without extra active operations; 3)highly scalable, which means the system can be deployed in multi-layer skyscraper like buildings. 4)least extra hardware deployment, for you cannot prospect users to actively equip with an extra gearing that is essential for your system, and the ideal scene is the only needed equipment is barely the smart phone. However, despite the strong demand, there are no existing systems that can cover the four characteristics listed above.

RF-based indoor localization systems such as RADAR delivers restricted accuracy. Indoors positioning system that rely on hardware such as RFID tags are restricted by the inconvenience and extra cost of hardware deployment. Existing visible light positioning systems as Luxapose have good performance in accuracy and with the only aid of smart phones, but it fails in the situation that it requires user to actively taking a photo and this is obviously inconvenient for user to follow, and besides, it's supported landmarks are strictly related to the performance of camera parameters, and for its experiment platform of Lumia 1020, no more than a hundred landmarks can be identified distinctly since its encoding algorithm is heavily dependent on the exposure time of smart phone camera. In section 2, we will identify these existing indoor positioning systems and try to list their flaws at the environment of actual deployment.

In this paper, we propose a new approach that can provided all the four criteria listed above. We follow the work based on AOA algorithm and rolling shutter effect of smart phone camera proposed by the work of Luxapose, and targeting its defects on encoding mechanism that can only identifies handful landmarks and user-friendly scenarios that need users to actively taking a picture to locate themselves, we give our solutions. In order to make the landmarks that are distinguishable to be scalable, we raised an encoding algorithm based on the idea of frequency domain multiplexing(FDM) by mixing the RGB channels of LEDs to convey adequate messages with a blink frequency of 4000 Hz. We applied manchester codes to our encoding mechanism so that the mixed LED light will be white to human eyes. Users can taking the picture at a high exposure time and low ISO value to exploit the rolling shutter effect, and the stripes on the picture can be processed by FFT knowing its frequency is 4000 Hz, then the mixed of RGB can be detached at Hue-Saturation-Lightness (HSL)cylindrical-coordinate so that we can obtain its encoding message. With the aid of web-based

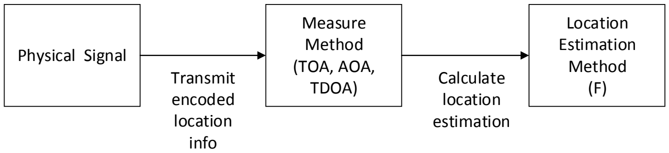


Fig. 1. Architecture of general IPS.

indoor maps, we can locate the user at a relatively close aera. If the user needs centimeter level precision, he/she can take a picture with multiple(no less than three) landmarks(led luminaries) and with the aid of AOA algorithm, the system can provide the result. We propose a scenario that users can obtain similar map experience indoors as outdoors that based on GPS module. In this scenario, we exploit the smart phone 3D acceleration sensor and compass sensor, we can obtain the user's track by software and with history based learning strategy we can remind users to adjust their location by our visible light positioning system to erase the deviation. So in this scenario, the indoor malls need to add a control module to the rgbleds, and the users need to install the smart phone app to work with the server, and all together we can fulfill this indoor positioning scenario.

II. Background and Motivation

Instantly knowing the location indoors is feasible in several related works. Specifically, there are two types of them: signal based IPS(indoor positioning system) and visible light based IPS. Signal based IPS, includes ultrasonic(US) and infrared(IR), and radio-frequency(RF) signal which may based on radio-frequency identification (RFID), received signal strength (RSS) of RF signals, Bluetooth wireless local area network (WLAN), ultra-wideband (UWB), etc[7]. Visible light based IPS usually exploits visible light with computer vision to work together as a method in IPS. It works with embeded CMOS cameras in smart phones which needs hardly additional facilities and provides fine results.

An indoor positioning system generally has an architecture of 3 phrases as shown in Figure 1. First step is to expose physical signals that contains the location information by US, IR, RF or other signals described above. These signal travels between emitter and receiver token by users. Then various methods are applied to calculate the physical quantity like measuring time of flight(TOF), time of arrival (TOA), time different of arrival (TDOA), angle of arrival (AOA), received signal strength (RSS) etc. With the raw information of a physical quantity measured, various techniques and algorithms are used which transform raw data into usable position information. Techniques have been classified as triangulation/trilateration [9], minmax algorithm [10], maximum likelihood[11] and fingerprinting [12].

RF systems estimate user location by measuring different properties of RF signals. RF Received Signal Strength Information (RSSI) has been employed for many different location estimation systems [13]. Performance of RSSI based location systems is highly effected by errors in signal strength resulted from multipath fading, reflection shadowing, diffraction etc [7]. Most of RSSI based IPS are exploiting WLAN as the media for physical signal since its low cost and convenience such as RADAR [2]. It uses signal strength information gathered at multiple receiver locations to triangulate the users coordinates. Triangulation is done using both empirically-determined and theoretically computed signal strength information. Because the drawbacks mentioned above on WLAN's multipath effect or so, it can only reach an estimation accuracy of 10 meters.

Ultrasonic based IPS provide fine-grained location with centimeter level accuracy and it have higher capacity of location to serve many users simultaneously. As the work of Cricket[5], it relies on TOF measurement of US signal calculated using velocity of sound and other signal combined such as RF signal. But unlike RF signals, velocity of sound in air does not remain constant and varies largely with environmental condition especially humidity and temperature [7]. For example, the speed of sound is:

$$v_{us} = 20.05\sqrt{T} \tag{1}$$

Which means for each kelvin at normal indoor environment, the deviation is close to 0.18%, an untolerated error. Another challenge to US systems is offered by high levels of environmental US noise. If noise is non-persistent, erroneous location estimates are filtered by use of suitable algorithms but persistent noise sources degrade system performance. Despite these challenges. deploying a US based IPS means a lot more hardware deployment work is to be done, and the users must be equipped with US receiver to get the system worked. This inconvenience can not be ignored in large-scale popularization.

RFID as a widely applicated low-cost hardware is also been developed as an method in IPS. Compared as RF and US signals, all RF tags can be read despite extreme environmental factors, such as snow, fog, ice, paint, and other visually and environmentally challenging conditions with a remarkable reading speed of less than 100 milliseconds. LANDMARC [8] is a RFID based IPS. It develops an algorithm to reflect the relations of signal strengths by power levels and maps which power level corresponds to what distance with the method of TDOA, and it reached a result of 2 meter level deviation. However it also have some problems. Currently available RFID products provides the signal strength of tags directly. Instead, the reader reports detectable or not detectable in a given range. This forces LANDMARC to spend approximately one minute each time to scan the 8 discrete power levels and to estimate the signal strength of tags which means the cost is unignorable. Besides, to apply this IPS, the mall needs to deploy several RFID tags and one RFID reader at an office-size area.

Apart from signal based IPS, there are works that relies on visible light to deliver the physical location information. Visible light are free of multipath, shadowing or attenuation of wireless signals, and are free from the influences of environmental temperature and moisture. One visible light based IPS, Epsilon [6], uses LED beacons and a custom light sensor that plugs into a smartphones audio port, and sometimes requires users to perform gestures. The LEDs transmit data using BFSK and avoid persistent collisions by random channel hopping. The system offers half-meter accuracy. This system requires custom hardware on user's phone and the performance can be improved. Luxapose [1] is another visible light based IPS which relies on the LED luminaries deployed on ceilings to deliver physical location information and exploits CMOS smart phone camera to do some image processing work to get the physical signals, and by AOA algorithm it estimates the location with an accuracy of several centimeters. However in this work, each LED luminary exploits blink frequency to transmit raw message, and exploit CMOS camera's rolling shutter effect to receive these raw message in token pictures, which means the camera on smart phones must support sufficient wide range of exposure time to receive distinct LED signal. In its FFT method of decode the rolling shutter image into encoded frequency and the experiment platform of Lumia 1020 camera, it can only support about 20 distinct led beacons with the exposure frequency of camera up to 16000 Hz and the necessarily barrier between distinct LED signals of 500 Hz, and this problem is crucial in deploying this IPS in practical indoor malls or airports. Another issue is that for each time the user want to track himself/herself, he/she must take out the phone and shoot a picture of the LED luminaries on the ceilings. Obviously for each time, locating oneself needs to take a picture is not a good kind of user experience and many applications based on the track of people such as automatically merchant recommendation can not be implemented under this IPS.

Table 1 is the benchmark comparison between the 5 IPS mentioned above.

param	RADAR	Cricket	LANDMARC	Epsilon	Luxapose

TABLE I
COMPARISON OF SEVERAL IPS

III. System Architecture

This system consists of RGBLED based visible light beacons, smartphone with camera and a processing server as shown in Fig 2.

We suppose an idea named **beacon source** as shown in the architecture Fig 2 of positioning beacon source 1 and positioning beacon source 2. In an indoor digital map, one distinct beacon source stands for one location coordinate on this map, and this coordinate is encoded in the light so if we can decode it we can roughly get our position nearby this coordinate. For each beacon source, the encoding data is provided by the controlling module which actually controls the flashing mechanism of each RGBLED bulb and we implemented it based on ArduinoDue board. In reality world, we can expolit the existing densely distributed RGBLED luminaries on ceilings of mall with only a low-cost PCB controller attached to its power supply to deploy our positioning hardware.

For each beacon source, it consists of no less than three RGBLED luminaries to support the angle-of-arrival algorithm.

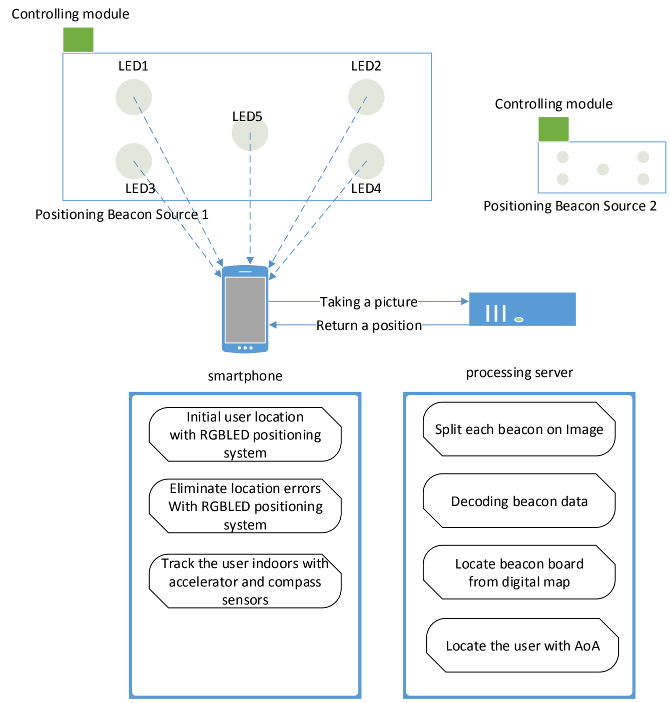


Fig. 2. Architecture of our system.

With knowing the roughly location with the help of beacon source and digital map, we can apply AoA algorithm so that we can calculate a relatively precise location of the user.

For each time the user wants to locate himself/herself, it is necessary to take a picture and post it to the remote server and get the location. Clearly it is not convenient. So we developed an extended smartphone application which works with 3D accelerator sensor and compass sensor which can roughly track the user's locus. The accumulated error can be erased by taking an RGBLED positioning action.

IV. Basic Design

Our system adpots an RGB mixed visible light as a communication channel to deliver locationing data. The RGBLED luminaries work as the transmitter and the smartphone camera works as the receiver. The encoding section describes how we encode digital map information into light signal with red, green and blue subchannels. The decoding section describes how we recover the message from mixed visible light. The positioning algorithm gives out detailed equations of angle of arrival that how can we measure the location based on all kinds of data we get.

A. Encoding

We deploy five RGBLED luminaries with their relative geometry known to provide redundancy in case some of them cannot be decoded. In fact under ideal circumstances three RGBLED luminaries will make the system locate the user's 3D location [14]. In other systems such as Luxapose [1] which exploits frequency as the method to distinguish different

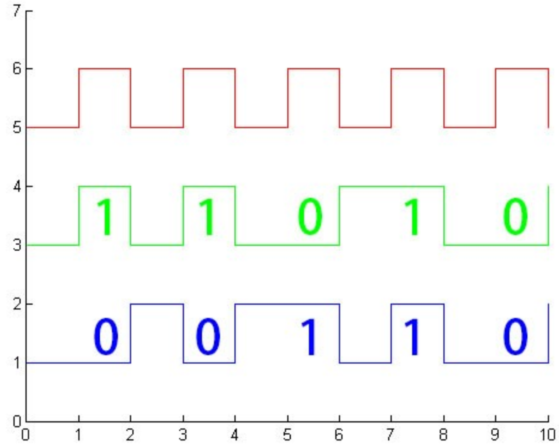


Fig. 3. Timing Diagram of Manchester Encoding.

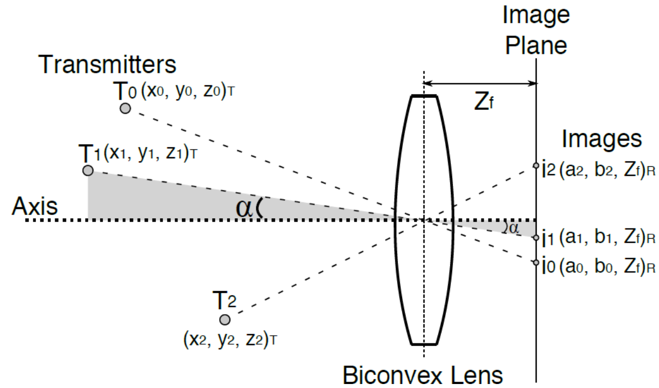


Fig. 4. AOA algorithm.

beacons, the major problem in practice is that it would need enough large barrier between the different frequencies so that the different stripes from rolling shutter effect could be distinguishable against the error casued by the photoing procedure. We introduce a novel encoding schema based on RGBLED with the method of frequency-domain multiplexing. By encoding different messages distinctly on red, green and blue channel with manchester coding which has a duty ratio of 50% at a high frequency such as 4000Hz in our experiment, the beam radiates as white to human eyes which means this light can be used as normal illumination indoors.

The three channels work at a same frequency. We select the red channel as the clock channel, which means the red led will be light and dark at a constraint frequency, and it can cut consequent time into fixed size small pieces. The green and blue leds will perform each single signal in each piece based on red channel. The green channel encodes the geographic information of the beacon source the LED belongs to. The blue channel encodes the identity information that can tell the LEDs on the same beacon source apart. Fig. 3 shows one optional encoding implement with a coding length of 8 bit. The encoding digit length is restraint to how many stripes can be discovered on one single light spot on image, and this depends on the distance between camera and RED light as well as the resolution of camera. For our experiment case with Lumia 1020 camera, we can decode the image at a range up tp 5 meters with a resolution of 15M pixels, and under this circumstances we can encode in 8-dight length which stands for 8-bit symbol. So we can get 256×256 distinct RGBLED luminary.

B. Decoding

First we need to find the centroid and size of each led spot on the rolling shutter image taken by the smartphone camera. The method is an image processing pipeline. First we need to rotate the picture to a standard manner since people take pictures in different angles. Then we convert the image to grayscale and we apply blur to it. After that we exploits a OTSU filter to find the edges of different spots on the image. We locates contours for different spots, and get the centroid by its circumcircle. With the centroid got, we know the spot maps to where on the image, and with the mask by the contour, we can respectively get each light spot for further processing.

We need to calculate the encoding frequency with the help of rolling shutter effect. We can compress the image subregion of light spot vertical to the rolling shutter stripes, and outcome a vector and take an FFT of it. By this way we can convert the time-domain stripes into frequency-domain result with an error less than 200Hz. By declaring that the transmit frequency be interger multiplies of five hundred, we can enlarge the channel up to $n/1000$ times with n stands for the reciprocal of exposure time parameter of smartphone camera, about 1/10000 second up to normal phones. For a smartphone with a camera of 1/10000 second exposure time, the expanding multiples of encoding channel is 10.

In our encoding schema, for each timeslice, we can get a RGB value in one of the eight circumstances of #000000, #0000FF, #00FF00, #00FFFFF, #FF0000, #FF00FF, #FFFF00, #FFFFFF,. The control unit can make it sure that each signal is strictly aligned in each timeslice. So the main idea is masking pure red to get timeslice gaps, and with the help of timeslice gaps, we use green and blue masks respectively to gain raw encoding information.

C. Positioning Algorithm

By decoding the green channel with the aid from server end digital map data, we can roughly locate ourselves around one certain beacon source. We adopt the arrival of angle algorithm to obtain precious positioning results. With a well-token image which means at least three distinct RGBLED luminaries can be decodable, we can determine the 3D position of the camera with respect to the beacon source.

As shown in Fig.4, the RGBLED luminary locate at point $T_0(x_0, y_0, z_0)_T$, this coordination is in the coordinate system on the positioning beacon source board and this is known to us but not the required coordination that we want to know. The luminary T_0 with the center of convex lens $(0, 0, 0)$ as well as the projection on the camera film $i_0(a_0, b_0, Z_f)_R$ stand in a line. Z_f is the distance between lens and film, and a_0, b_0 can be got by locating the spot on the image as we talked above in section 4.2. Obvirously with geometry laws we can get the coordination of $T_0(u_0, v_0, w_0)$ that relative to the lens:

$$\begin{aligned} u_0 &= K_0 \times a_0 \\ v_0 &= K_0 \times b_0 \\ w_0 &= K_0 \times Z_f \end{aligned} \tag{2}$$

Here K_0 is an unknown. Obvirously, the goal is to calculate $T_0(u_0, v_0, w_0)$. We have three pairs of this equation for three LED luminaries. And we know the location of $T_0(x_0, y_0, z_0)_T$, $T_1(x_1, y_1, z_1)_T$ and $T_2(x_2, y_2, z_2)_T$. Since we know the precise deploying location of T_0, T_1 and T_2, we can thus obtain the precise location of the user.

We define d_0 as the distance between T_0 and T_1, we can get the equation below:

$$\begin{aligned} d_0^2 &= (u_0 - u_1)^2 + (v_0 - v_1)^2 + (w_0 - w_1)^2 \\ &= (K_0 a_0 - K_1 a_1)^2 + (K_0 b_0 - K_1 b_1)^2 + (K_0 c_0 - K_1 c_1)^2 \\ &= K_0^2 [\vec{O_{l_0}}]^2 + K_1^2 [\vec{O_{l_1}}]^2 - 2K_0 K_1 (\vec{O_{l_0}} \vec{O_{l_1}}) \end{aligned} \tag{3}$$

Here $\vec{O_{l_0}}$ and $\vec{O_{l_1}}$ stand for a vector begin from the center of convex lens and end at $i_0(a_0, b_0, Z_f)_R$ and $i_1(a_1, b_1, Z_f)_R$. K_0 and K_1 are unknown to us. This is only two of the three LEDs, and by exploiting all three LEDs we can get a ternary quadratic group which means we can solve K_0, K_1 and K_2 out. With K_0, we can get the location of C with the help of equation[2]. Similarly we can get $T_1(u_1, v_1, w_1)$ and $T_2(u_2, v_2, w_2)$.

However, this is an ideal scenario. Actually, the procedure when we locate the spot on image to get the $i_0(a_0, b_0, Z_f)_R$, we may generate cumullative errors. In order to reduce the errors, we transform the known coordinates of $T_0(x_0, y_0, z_0)_T$, $T_1(x_1, y_1, z_1)_T$ and $T_2(x_2, y_2, z_2)_T$ into an optimization problem. We will seek the minimum square error by a series quantization of parameters as in equation 4. Here N stands for the total number of distinct RGBLED luminaries, so if we deploy N LEDs on one beacon source, we get $\binom{N}{2}$ equations.

$$\begin{aligned} \sum_{m=1}^{N-1} \sum_{n=m+1}^{N} &\{K_m^2 [\vec{O_{l_0}}]^2 + K_n^2 [\vec{O_{l_1}}]^2 \\ &- 2K_m K_n ([\vec{O_{l_m}}][\vec{O_{l_n}}]) - d_{mn}^2\}^2 \end{aligned} \tag{4}$$

As long as all quantized parameters are estimated, the locations of light sources in the camera receiver coordination are fixed. Then the position of the receiver against the beacon source can be transformed as equation 5 below.

$$\begin{bmatrix} x_0 & x_1 & \dots & x_{N-1} \\ y_0 & y_1 & \dots & y_{N-1} \\ z_0 & z_1 & \dots & z_{N-1} \end{bmatrix} = R \times \begin{bmatrix} u_0 & u_1 & \dots & u_{N-1} \\ v_0 & v_1 & \dots & v_{N-1} \\ w_0 & w_1 & \dots & w_{N-1} \end{bmatrix} + T \tag{5}$$

Here R is a 3×3 rotation matrix, T is a 3×1 transfer matrix. The vector $T(T_x, T_y, T_z)_T$ stands for the coordinate of camera signal receiver in the coordination of LED luminaries. With the quantized parameters known, and the relative coordination between the LED signal transmitter and camera receiver is fixed, we can get the coordinate of camera signal receiver at the coordination of LED luminaries as shown in equation 6 below:

$$\begin{aligned} &(T_x - x_m)^2 + (T_y - y_m)^2 + (T_z - z_m)^2 \\ &= K_m^2 (a_m^2 + b_m^2 + Z_f^2) \end{aligned} \tag{6}$$

Here (x_m, y_m, z_m) is the mth light source's coordinate on the coordination of beacon source, (a_m, b_m) is the coordinate of the corresponding light spot on the camera film coordination. We will give a prediction of vector $T(T_x, T_y, T_z)_T$ to minimize the value of equation 7 below:

$$\begin{aligned} \sum_{m=1}^{N} &\{(T_x - x_m)^2 + (T_y - y_m)^2 + (T_z - z_m)^2 \\ &- K_m^2 (a_m^2 + b_m^2 + Z_f^2)\}^2 \end{aligned} \tag{7}$$

As long as we determine transfer vector $T(T_x, T_y, T_z)_T$, we can identify each unknown in rotation matrix R as:

$$R = [\vec{r_1}, \vec{r_2}, \vec{r_3}] \tag{8}$$

V. Implementation

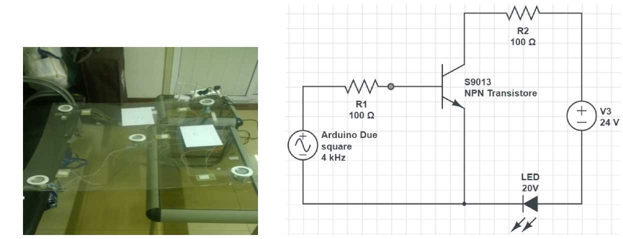
We implement a prototype system in order to check the performance of our system. And we develop one additional Android application to enrich the scenario in actual circumstances by exploiting its 3D accelerometer and compass sensors to record the user's track. So the final scenario is the user can track him/herself with the aid of smartphone apps, and eliminate the accumulated error at the prompt of the app by analysis of historical record with high accuracy visible light positioning. This is because using visible light positioning alone requires the user to actively take an image of one positioning beacon source. This is inconvenient if a user want to locate him/herself as long as he/she is indoors, and many potential applications such as automatic recommendation of merchants by location infomation would be inaccessible.

A. *Encoded RGBLED Board*

We build one beacon source using a square plexiglass board as a carrier plane for five distinct RGBLED luminaries. As is shown in fig.5_a, the luminaries are set at the four corners and the center of the square board. We attached one Arduino Due board at the back side and programmed our encoding into it as the controller.

For each distinct RGBLED transmitter, we put it horizonly at the bottom of a lamp tube with a piece of frosted glass cover the tube in order to scattering the transmitting light into soft light.

The amplifier circuit is designed as in fig.5_b. In fig.5_b, it presents an amplifier circuit for one channel of a single RGBLED. We exploit the S9013 NPN transistor to work as a micro relay from the 3.3V encoding circuit and the 24V DC power supply. The controlling signal varies at a 4000Hz frequency, when the signal is high, the transistor works as wire



(a) Positioning Beacon Source board. (b) Amplifier circuit.

Fig. 5. Hardware implementation

Fig. 6. Pipeline to locate spots.

and the RGBLED is bright, and when the signal is low the transistor is disconnect so the RGBLED would be dark.

The encoding controller is the output of one digital port of Arduino Due board, and we extend 15 outputs from a single board to control 5×3 channel signals with our encoding schema in section 4.1.

We use the Nokia Lumia 1020 smartphone with Nokia Camera Explorer to take photos. It has a high resolution of 7712×5360 pixels. With known locations of camera and the positioning system set, we take an experiment and get the Z_f parameter as 5470 pixels.

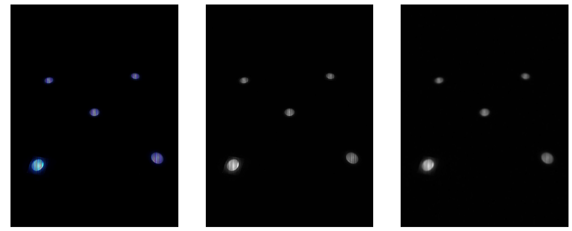
B. Decoding and Positioning Server

As the image processing procedure would be executed with high overhead, we perform the processing procedure in the server by a restful operation of posting the image to the server and take back the location results. The server is responsible for decoding and calculating the AoA algorithm. In order to decode the image, we should first find the pixel location of the light spots on the image as described in section 4.2.

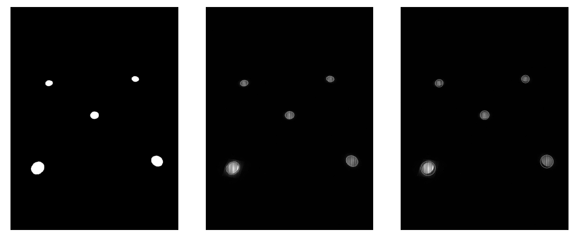
*1) **Locating light spot on image**:*

The process can be diveded into a series of specific operations as shown in fig.6. We use C++ based OpenCV 2.4.11 to perform such processings. The processing pipeline takes 3 seconds to perform one entire process and can undertake at least 20 concurrently on a server with a CPU of Intel i5-5200U and 8G memory. After we take a rolling shutter picture, we get an image like fig.7_1.

In order to locate the light spot, first we transfer the origin image into **grayscale**. Since the average grayscale value of light spot area significantly differs from dark regions, we can have an accurate recognition of the light spots on the image. Since we have performed high frequency and thus results in rolling shutter effect, the light spot are composed of disconnected stripes rather than a continous light spot. In order to locate the centroid of the spot, we need to blur it into a continous region. We perform a **Gaussian Blur**, that is the



(a) Original image. (b) Gray into single channel. (c) Blur to find rough location



(d) Get spots' location mask. (e) Find contours (f) Plot the circumscibed circle

Fig. 7. Locating spot on image

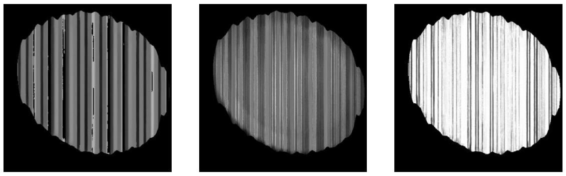
pixel value after the blur process is the average of a circle region area before the process, and we get a blurred image like fig.7_c. We need to find a mask board that can identify these light spots individually, so we perform a **OTSU filter** to find the threshold and generates a mask like fig.7_d. The threshold is set to 0.5, and those pixels with a grayscale value smaller than 0.5 will be replaced by 0, with those bigger than 0.5 be replaced by 1.0 grayscale as the white part in fig.7_d.

We will use the grayscale border to construct the **contour** for each light spot as in fig.7_e. In order to perform the AoA algorithm we need to know the coordinate of each light spot on image. We find the circumscribed circle for each contour, and adopt its **centeroid** as the coordinate.

*2) **Decode the infomation**:*

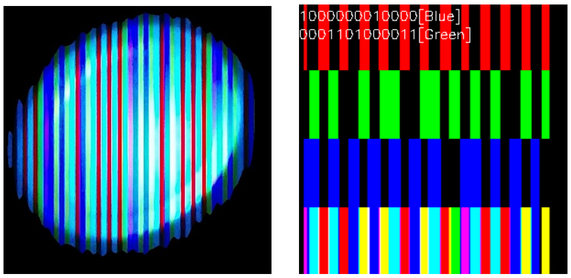
We decode the information hidden in the rolling shutter effect in order to find the Green channel information of where the positioning beacon source locates indoors, and with the Blue channel to match a light spot with the corresponding RGBLED luminary on the board so as to perform the AoA algorithm afterwards.

Ideally, the encoding information should have only the difference on the Hue axis of the HSL cylindrical-coordinate with same Saturation and Lightness. However, in the taking picture process, other light source may introduce error on the token picture that results in errors on saturation and lightness. Suppose a pixel $p(r, g, b)$ with r,g,$b \in [0, 1]$, and max is the biggest one among r, g, b, min is the smallest one among r, g, b. We prform a conversion from RGB space to HSL space, and separate the HSL channels as in fig.8 as in the equation 9∼11 below:



(a) Hue channel. (b) Saturation channel. (c) Lightness channel

Fig. 8. One spot on HSL cylindrical-coordinate



(a) Original image. (b) RGB channel decodes.

Fig. 9. Retrieve RGB infomation from one spot

$$h = \begin{cases} 0^\circ & if \quad max = min \\ 60^\circ \times \frac{g-b}{max-min} + 0^\circ & if \quad max = r \quad and \quad g \geq b \\ 60^\circ \times \frac{g-b}{max-min} + 360^\circ & if \quad max = r \quad and \quad g < b \\ 60^\circ \times \frac{b-r}{max-min} + 120^\circ & if \quad max = g \\ 60^\circ \times \frac{r-g}{max-min} + 240^\circ & if \quad max = b \end{cases} \tag{9}$$

$$s = \begin{cases} 0 & if \quad l = 0 \quad or \quad max = min \\ \frac{max-min}{max+min} = \frac{max-min}{2l} & if \quad 0 < l < \frac{1}{2} \end{cases} \tag{10}$$

$$l = \frac{1}{2}(max + min) \tag{11}$$

Here $h \in [0, 360)$, $s, l \in [0, 1]$. Since for each stripe, it stands for the state of one RGBLED at a certain time. In order to get stable encoding transmission, we compress the three images of H,S,L into row vectors to cut down errors. By this operation, we can locate the most close Hue, Saturation and Lightness and the performance is nearly 15% percent better than just compress them in RGB space since by that way the ambient light can cause effect on Hue, which would raise error since our encoding is purely Hue based.

After the process of cutting down errors, we restore the HSL image back to RGB space, and at this time it shows like the bottom row of the four rows in fig.9_b compare to fig.9_a before this process. We perform a red mask to get the red clock channel, and with clock information and green, blue masks, we recover the RGB three channel encodings as the top three rows in fig.9_b.

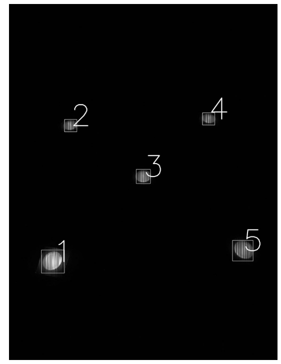


Fig.10 Output after processing.

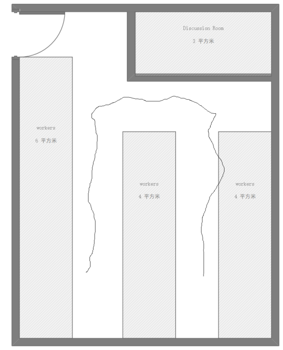


Fig.11 User Tracker

After the total image processing, we get a tagged output image as fig.10. The digit on each light spot shows the corresponding relationship between the light spot on image and the specific RGBLED luminary on the positioning beacon board shown at fig.2.

Then first we use the green channel information to locate the positioning beacon source location based on indoor map database. Then as we have gathered all parameters, we can perform the AoA algorithm to locate the user precisely.

C. *Smart Phone Tracker*

We extend the smartphone app to reach a goal that can **automatically** track the user and support *Location Based Services* since by the visible light positioning system alone, for each time the user wants to locate him/herself, it would be necessary for him/her to perform a taking-photo action, and it would be impossible for services to locate the user so LBS can not be supported. In order to fulfill this demand, we exploit the smartphone's 3D accelerometer and compass sensors and developed an Android application that can support it. By 3D accelerometer, we can count the user's step length and by the compass, we can record the direction of movement of a user. Combine them and we can determine the user's track as in fig.11.

Modern smartphones are equipped with a 3D accelerometer, it can record the acceleration value of the x, y and z axes. In reality environment, the status of a smartphone would be random in the user's pocket. As in fig.12, it shows a unit cycle of walking behavior with changes in vertical and horizontal acceleration. So we can always sum the three acceleration vectors and find two peaks which are perpendicular to each other, and we can identify the vertical and horizontal one by the norm of the vector, with the larger one as the forward vector.

In order to find the correct peak from the interference of handshake made accelerator changes, we will record a historical recorded threshold and step frequency as a filter. Only the unconventional peaks can be recorded and analyzed.

We also exploits the compass sensor to record the track orientation of the user. With a pedometer and moving direction, we can track the user at a rough range. As is mentioned in [16], this kind of tracker has an error of 5% $\sim$ 15% in direction and

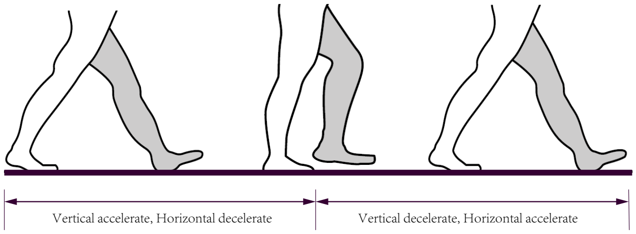


Fig.12 Walking stages and acceleration pattern.

$3\% \sim 5\%$ in pedometer. So we need to introduce a remind mechanism to inform the user using visible light positioning system to erase the accumulated error. Based on the historical dataset, users are recommended to take a picture and carry out a visible light positioning to make sure the deviation within an acceptable range set by the user.

VI. Performance and Evaluation

We deploy our experiment motherboard in a grid divided $10m \times 10m$ office room, and the board is a $1m \times 1m$ square shaped board. We performed the experiment to measure the accuracy quota under different imaging **distance**, **angle**, **brightness levels**, and **encoding frequencies**. The main benchmark is accuracy. The power of the LEDs are 20W with a 24V voltage on green and blue LEDs and 20V on red LED since they works at distinct voltages. We put the RGBLED in a cylindrical lampshade and covered a piece of frosted glass to make the light source scattering uniformly. The RGBLED light source is a $2.8cm^2$ square shaped LED board, and we placed five RGBLEDs on a board, one at the center, and four at the corners. We verify the location error by the absolute value of the visible light positioning system calculation results and the measurements with a band tape which can accurate to the centimeter.

A. *Realistic Positioning Range*

We set up the test board on the corner of office, vertical to the ground, and perform the experiment at different distance beyond the board with a $0°$ imaging orientation. For a distance between user and board at 10m, the decoding program starts to report exceptions that it failed to discover light beacons on the image, so under the specific hardware environment, the effective application distance is 10m.

We perform 25 tests independently for each distance of 2m, 4m, 6m and 8m, and the CDF chart of these tests is shown as fig.13(a). Obviously distance makes different accuracy results and the closer the user taking an image, the better accuracy he/she gets. At a 2m distance which is usually larger than the ceiling height minus human height, this system can reach an accuracy within 20cm for 80% of all cases. And even in a range of 8m, user can locate him/herself within half a meter, which is meaningful in most indoor applications.

B. *Maxim Imaging Angle*

In this evaluation, we perform the test at a distance of 2 meters, and taking images at different angles to verify how angular affects accuracy. The board is set vertical to the ground. We find that for an imaging angle larger than $0°$, the decoding program starts to report exceptions that it failed to discover light beacons on the image, so under the specific hardware environment, the effective angle to take the positioning work is within $60°$

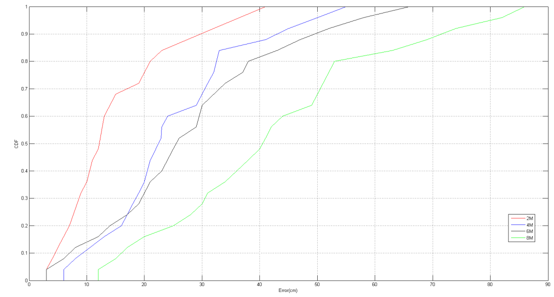


Fig.13(a) Distance error.

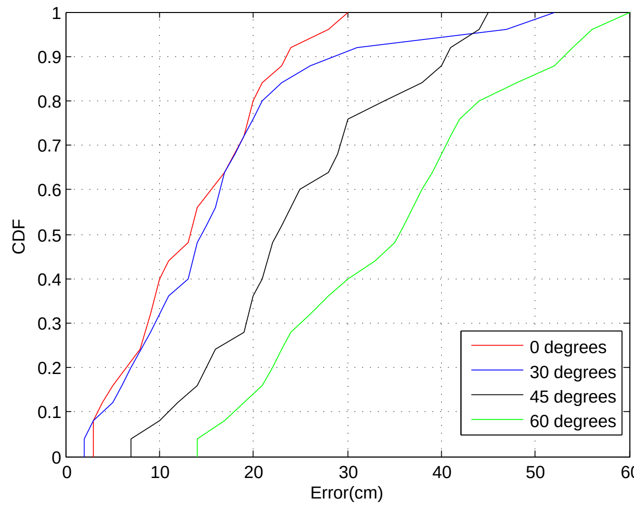


Fig.13(b) Angle error.

We perform 25 tests independently for each imaging angle at $0°$, $30°$, $45°$ and $60°$, and the CDF chart of these tests is shown as fig.13(b). From the chart we can identify that at small imaging degrees, the positioning system worked as well as there exists no angle difference, and this system can reach an accuracy within 20cm for 80% of all cases. However, when angle grows larger, it also affects the accuracy. For a $45°$ circumstance the error grows to 33cm for 80% cases, and at $60°$, the error is nearly 45cm, with several exceptions that result in nothing not counted. This is because with the growth of angle, the RGBLED spots on the image distorted and this caused the image processing pipeline exception.

C. *Environment Brightness Effects*

We perform this experiment under different ambient brightnesses at 2m and $0°$. In order to control the brightness, we perform the experiment at night and fabricate 3 level brightness with no light(dark), slight light with lamp on ceiling bright and

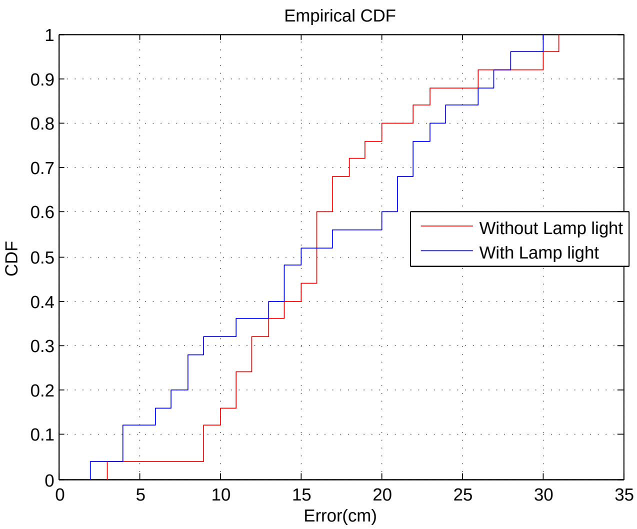


Fig.13(c) Lamp lighting's influence.

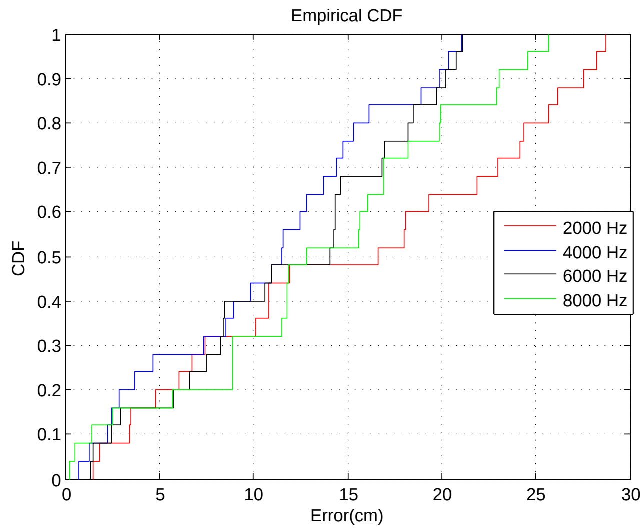


Fig.13(d)CDF at different encoding frequency.

high lightness with experimenting at daytime, and the CDF chart of these tests is shown as fig.13(c). We have to admit that at daytime with heavy environmental noise brightness, the positioning system can't decode the polluted RGB light signal well which results in failure at taking a decodable rolling shutter image using normal smartphone cameras such as Lumia 1020 . However for normal indoor environment, the visible light positioning system itself works as the light source, and it suffers little influence from weak nearby lamp light.

D. Encoding Frequency Effects

In theory, the encoded frequency of light source can be bounded within $\frac{f}{2}$ with f as the largest shutter frequency(reciprocal of exposure time, for Lumia 1020 this is limited within 1/16666 second) for the smartphone camera. In the experiments before we worked at 4000Hz, and in this section we adopt same encoding(8 bit length) and evaluate how encoding frequency influences accuracy. We set the imaging positioning 2m from the light source with 0° direction and without lamp light besides. We performs 2000Hz, 4000Hz, 6000Hz and 8000Hz experiment, and we see they worked as well in the cdf chart of fig13(d). This can confirm that within decodable frequency boundary, we can adopt frequency as a dimension to enlarge the coding space.

REFERENCES

[1] Ye-Sheng Kuo, Pat Pannuto, Ko-Jen Hsiao and Prabal Dutta. Luxapose: Indoor positioning with mobile phones and visible light. In Proc. 20th annual international conference on Mobile computing and networking (Mobicom 14) Pages 299-302.
[2] P. Bahl and V. N. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In Proc. 19th Annual Joint Conference of the IEEE Computer and Communications Societies. (INFOCOM 00), volume 2, 2000.
[3] C. Danakis, M. Afgani, G. Povey, I. Underwood, and H. Haas, Using a CMOS camera sensor for visible light communication, in Proceedings of the IEEE Workshop on Optical and Wireless Communication (OWC), pp. 1244- 1248, 2012.
[4] K. Chintalapudi, A. Padmanabha Iyer, and V. N. Padmanabhan. Indoor localization without the pain. In Proc. of the 16th ACM Annual International Conference on Mobile Computing and Networking (MobiCom 10), 2010.
[5] Nissanka Bodhi Priyantha, Hari Balakrishnan, Erik Demaine, Seth Teller, Mobile-Assisted Localization in Wireless Sensor Networks, Proc. IEEE INFOCOM Conference, March 2005.
[6] Liqun Li, Pan Hu, Guobin Shen, Chunyi Peng, Feng Zhao, Epsilon: A Visible Light Based Positioning System, the 11th USENIX Symposium on Networked Systems Design and Implementation, Seattle, Washington, 2014.
[7] Faheem Ijaz, Hee Kwon Yang, Arbab Waheed Ahmad, Chankil Lee. Indoor positioning: A review of indoor ultrasonic positioning systems, Advanced Communication Technology (ICACT).
[8] LIONEL M. NI and YUNHAO LIU, LANDMARC: Indoor Location Sensing Using Active RFID, Pervasive Computing and Communications, 2003. (PerCom 2003).
[9] J. Hightower, SpotON: An Indoor 3D Location Sensing Technology Based on RF Signal Strength, in: Department of Computer Science and Engineering, University of Washington, Seattle, WA, 2000.
[10] K. Langendoen, N. Reijers, Distributed localization in wireless sensor networks: a quantitative comparison, Computer Networks 43 (2003) 499518.
[11] M. Sugano, T. Kawazoe, Y. Ohta, M. Murata, Indoor Localization System Using RSSI Measurement of Wireless Sensor Network Based on ZigBee Standard, in: The IASTED International Conference on Wireless Sensor Networks (WSN 2006) Banff, Canada, 2006.
[12] K. Kaemarungsi and P. Krishnamurthy, Properties of indoor received signal strength for WLAN location fingerprinting, The First Annual InternatIOnal Conference on Mobtle and UbiqUitous Systems: Networking and Services, 2004. MOBIQU1TOUS 2004.
[13] E. Lau, ENHANCED RSSI-BASED HIGH ACCURACY REALTIME USER LOCATION TRACKING SYSTEM FOR INDOOR AND OUTDOOR ENVIRONMENTS International Journal, vol. 1, no. 2, 2008.
[14] Se-Hoon Yang, Hyun-Seung Kim, Three-Dimensional Visible Light Indoor Localization Using AOA and RSS With Multiple Optical Receivers, Journal of Lightwave Technology(Volume:32 , Issue:14),2480 - 2485.
[15] Neil Zhao, Full-Featured Pedometer Design Realized with 3-Axis Digital Accelerometer.