

Article

# An Integrated Object Tracking and Covert Visual MIMO Communication Service for Museum Security System Using Single Vision Sensor

C. Manikandan <sup>\*</sup>, S. Rakesh Kumar , K. Sai Siva Satwik , P. Neelamegam, K. Narasimhan  and N. Raju

School of Electrical & Electronics Engineering, SASTRA Deemed University, Thanjavur 613 401, India; srakesh@eie.sastra.edu (S.R.K.); saisivasatwik9@gmail.com (K.S.S.S.); neelkeer@gmail.com (P.N.); knr@ece.sastra.edu (K.N.); raju@ece.sastra.edu (N.R.)

\* Correspondence: manikandan@ece.sastra.edu; Tel.: +91-988-430-5589

Received: 15 September 2018; Accepted: 5 October 2018; Published: 15 October 2018



**Featured Application:** Single vision sensor (CCTV camera) is used for both object tracking and secured communication without requiring any additional hardware. IWT-based ARC-LSB techniques are employed for robust covert Visual MIMO communication. A Kalman filter algorithm is used for object monitoring. Apart from the museum, the proposed system can be applied in various public gathering locations such as railway stations, airports, shopping malls, and historical places. The proposed system can be implemented without any modification to the existing hardware architecture; this makes the proposed system cost-effective and less susceptible to circumvention.

**Abstract:** This paper explores the spatial coverage potential of a vision sensor to implement a dual service for a museum security system. Services include object monitoring and invisible visual Multiple-Input and Multiple-Output (MIMO) communication. For object monitoring, the Kalman filter algorithm is used. To implement a secret visual MIMO communication, an Adaptive Row-Column (ARC)-based LSB substitution technique is used along with the Integer Wavelet Transform method. These proposed services are implemented using existing electronic visual displays and surveillance cameras. Experiments are carried out using a sample object and four stego images. The performance of the object tracking system is measured in terms of accuracy by comparing the actual and estimated position of an object. Similarly, the performance of the covert visual MIMO system is measured at two different stages. On the transmitter side, differences between the cover and stego-images are used to measure imperceptibility. Likewise, at the receiver, differences between the original and the reconstructed data images are used to measure the accuracy. Results illustrate that the proposed system has improved imperceptibility, accuracy, and robustness, as compared to existing techniques.

**Keywords:** vision sensor; IWT; ARC; Hough line detection; Kalman filter; visual MIMO communication; object detection

---

## 1. Introduction

Nowadays, in every digital museum, inside the object showcase room, an electronic visual display and surveillance camera were fixed for each precious object. These display screens were used to present the history of an object in the form of moving images which will improve the quality of visitors' experience [1–5]. Similarly, surveillance cameras were used for monitoring the objects [6] and the

visitors' movement [7,8]. Due to the advancement and advantages of Optical Wireless Communication (OWC), Optical Camera Communication (OCC) have been used in many locations [9]. A primary OWC system consists of a single LED and photodiode, with the combination of the LED and photodiode acting as transmitter and receiver. The data rate of such a system depends on the reception rate of the photodiode [10]. Later on, the MIMO concept was extended to OWC systems [11]. In the OWC system, to implement the MIMO concept, LEDs are arranged in a matrix form called a Light Emitting Array (LEA). Each LED in an LEA is considered as a transmitting antenna, transmitting useful information in parallel. The LEA is captured using a camera and a region of each LED is extracted. The extracted region of the LED is considered as a receiving antenna. Hence, the system is called a *visual MIMO communication system* [12–14]. The visual MIMO communication system has found wide applications in outdoor settings such as Vehicle to Vehicle communications (V2V), Object tracking, and so on [15–17]. It is also tested in indoor applications [18,19]. Recently the authors proposed a novel Visual MIMO interface for interfacing computer and control systems [20]. This technique was further implemented as a screen-camera communication device by considering each pixel in the display and camera as a transmitting and receiving antenna [21]. To achieve hidden communication between the display and camera, various time and transform domain image steganographic methods were introduced [22–25]. From the captured images, specifically ROI, having few straight lines, were detected by Hough transform (HT) [26]. Similarly, for object detection, nowadays more efficient filters are used [27].

The significant contribution of this paper is (i) the realization of a dual security service using a single vision sensor; (ii) the use of a Kalman filter algorithm for object monitoring; (iii) the extraction of a display screen using Hough transforms; (iv) the implementation of a robust invisible visual MIMO communication system using IWT based ARC-LSB substitution technique; and (v) investigation into the performance of the proposed system using standard measurement parameters.

This article has been organized as follows. Section 2 presents the proposed system model. Section 3 investigates & compares the performance of the proposed algorithm against the existing algorithm. Finally, Section 4 concludes the paper.

## 2. System Model

The schematic representation of the proposed system is shown in Figure 1. The proposed system consists of a centralized monitoring-cum control room and object showcase room. From the centralized control room, the precious object presented in the object showcase room is continuously monitored by object detection and the tracking system, in that a Kalman filter algorithm is used to detect and track its location. The detection results obtained from the Kalman filter algorithm are used to control the alarm system. The same results are given to the computer to generate command information, which is transmitted as a stego image to the object protection system through a proposed closed-loop covert visual MIMO communication system. In this closed-loop security system, the surveillance camera present inside the showcase room captures the display screen along with the precious object. From the captured image, display screen region is given to the data image extraction algorithm, while the object portion of the image is fed to the object detection and tracking system. In the proposed covert visual MIMO communication system, the color cover images that contain information about the history of the precious object are used to embed the data images. Each color cover image consists of three planes, i.e., red, green, and blue, to embed the data images. After embedding, these three planes are combined to form a colour stego image, as shown in Figure 2. The combined stego images are given to the display screen. In this display screen, each  $8 \times 8$  pixel data block hidden in the stego image is considered as a transmitting antenna. The public visiting the object showcase room can see only the cover images displayed on the screen. They cannot visualize it as an array of transmitting antennas and embedded information. The surveillance camera present inside the showcase room captures the display screen along with the precious object. Inside the object showcase room, various image processing techniques are used to segment the display screen from the captured image. The segmented display screen is

considered as the received stego image. In the received stego images, each recovered  $8 \times 8$  pixel data block is considered as a receiving antenna. Various processes that happen in the proposed system are presented in a detailed manner in the subsequent subsection.

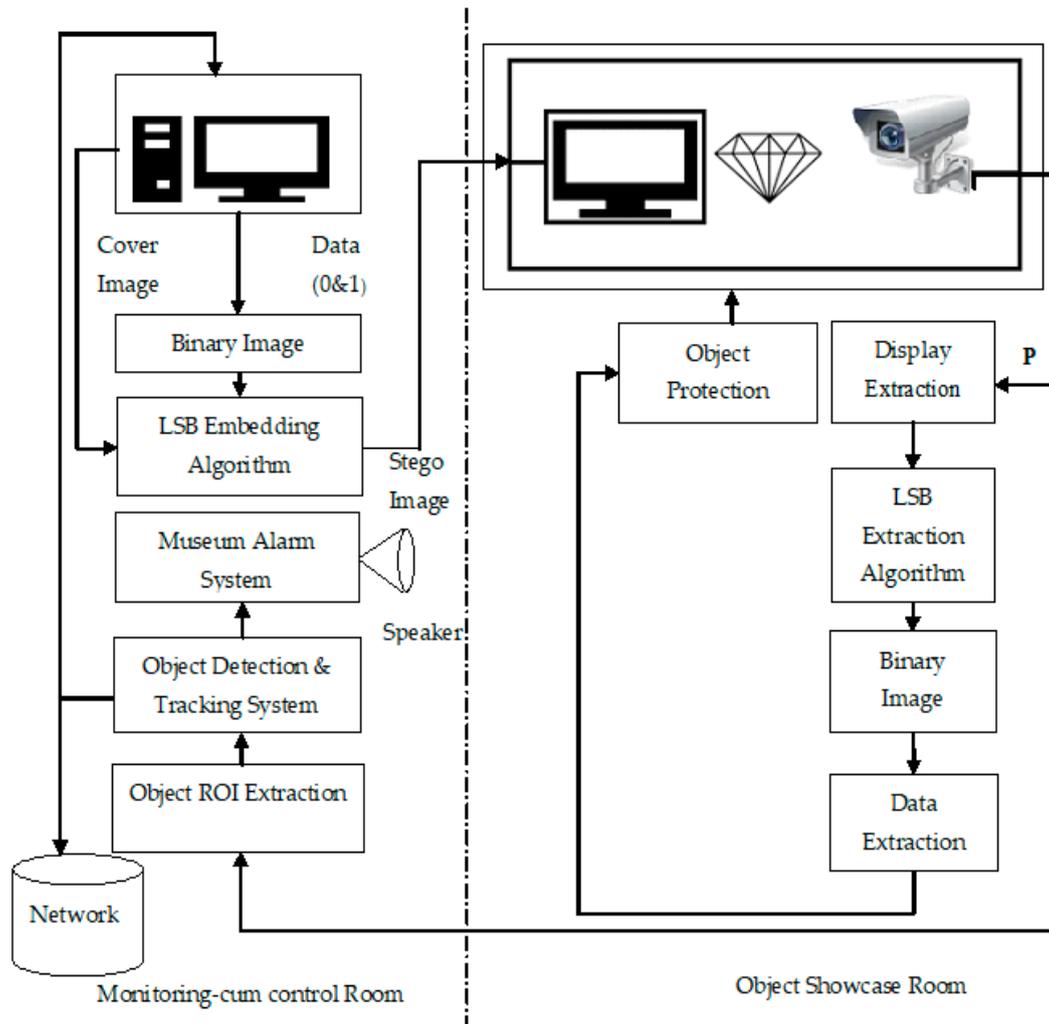


Figure 1. Block diagram representation of a proposed system.

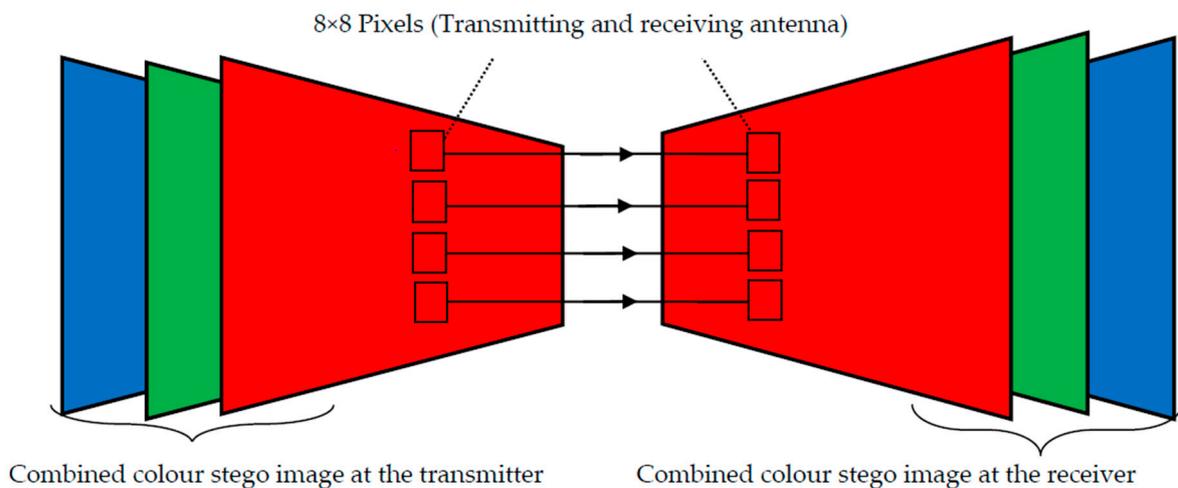


Figure 2. Screen-Camera based visual MIMO Communication Illustration.

### 2.1. Data Image Creation

In the control room, the command information to be sent is generated. This information is converted into a binary data stream and stored in an array. The following steps are used to create a black and white binary data image from the incoming data bit stream.

Step 1: Create a binary image of size  $1920 \times 1080$  by assigning '0' value to all the pixels. This obtained image is called a *black binary Image*.

Step 2: The black binary image is divided into  $8 \times 8$  blocks.

Step 3: Select the first  $8 \times 8$  block (containing 64 pixels).

Step 4: If the incoming data bit is '1', replace all 64-pixel values in that block by '1'. Otherwise, the pixel values in that block are unchanged.

Step 5: Select the next block in the same row and repeat Step 3 for that block. If the previous block is the last block in that row, then the next block is the first block in next row.

Step 6: Repeat Step 4 until every  $8 \times 8$  block in the black binary image is modified based on the incoming binary data bit stream.

Step 7: This process generates a black and white binary data image.

These black and white binary data images need to be embedded within the cover image. Since the cover image consists of three planes, Red, Green, and Blue, the above process is repeated for three images, as shown in Figure 3. It is important to note that the created black and white image and the cover image must be of the same size. The Integer Wavelet Transform (IWT) is applied to the created black and white image; this gives the Low-Low (LL), Low-High (LH), High-Low (HL), and High-High (HH) subbands. The LL subband ( $960 \times 540$ ) is selected; this LL subband of these three black and white binary data images is given to three different pixel coefficient readers, as shown in Figure 3. The pixel coefficient reader reads coefficients of the LL subband of black and white binary data images row-wise, from top to bottom. The output of these three-pixel coefficient readers is noted as A, B, and C respectively. The A, B, and C are binary bit streams created from the LL subband of the black and white binary data image.

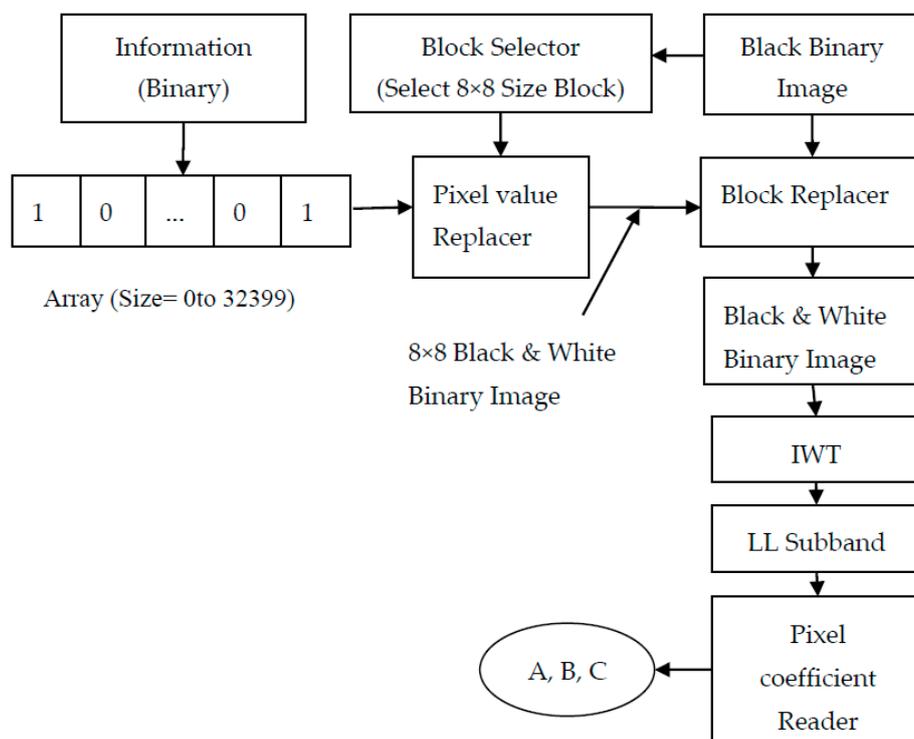
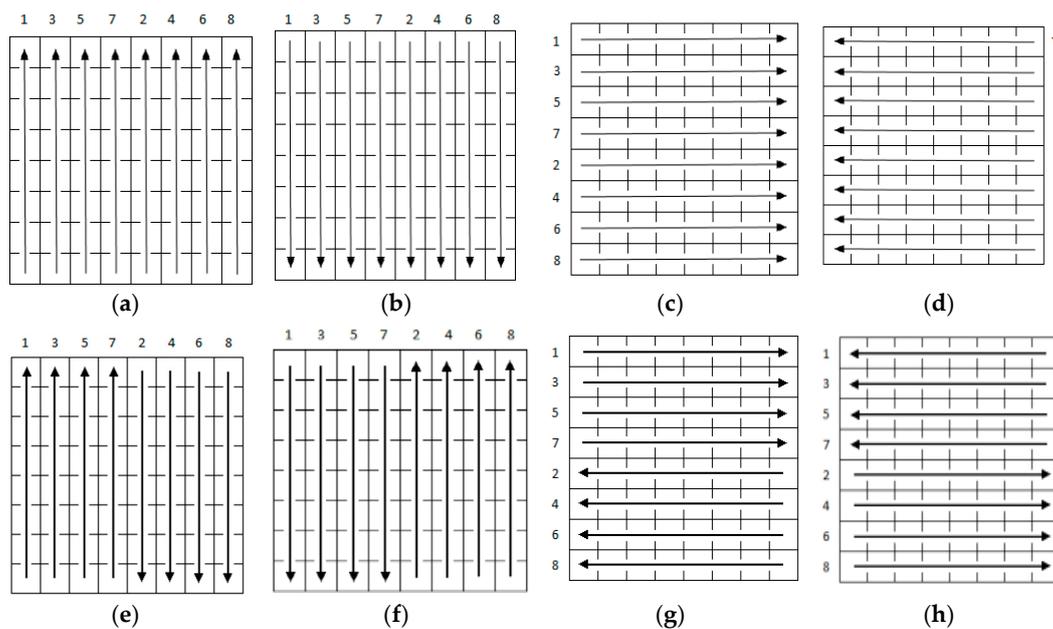


Figure 3. Black & White Binary Image Creation for Red, Green & Blue plane.

### 2.2. Embedding Algorithm

Four different cover images of size  $1920 \times 1080$  are selected. These cover images are color and contain information about the precious object. Each cover image consists of three planes, i.e., Red, Green, and Blue. Thus, the A, B, and C outputs obtained from the previous step are embedded into the HH subband of the cover image using following the steps:

- Step 1: Read the color cover image ( $1920 \times 1080$ ).
- Step 2: Divide the color image into Red (R), Green (G) and Blue (B) Planes.
- Step 3: Apply 2D-Haar IWT to R-plane of the cover image and generate LL, LH, HL and HH band.
- Step 4: Select the HH band in R-plane of the cover image.
- Step 5: Divide HH band into  $4 \times 4$  blocks and select one block at a time randomly, using a random number generator.
- Step 6: Read the black and white binary data image ( $1920 \times 1080$ ).
- Step 7: Apply IWT. This gives LL, LH, HL and HH sub-bands. Select LL band.
- Step 8: Replace LSB of each integer coefficient in the HH band of Red plane with LSB of corresponding integer coefficients in the LL sub-band of the binary data image1 using 8 patterns [24] as shown in Figure 4a-h.
- Step 9: Calculate Mean Square Error of each pattern and apply pattern with Least MSE for that block.
- Step 10: Select 3-bit key for the applied pattern from Table 1 and shift the selected key to left by three bits.
- Step 11: Repeat from Step 8 to Step 10 for the selected block until all data blocks of LL sub-band are embedded within HH sub-band of cover image in R-plane.
- Step 12: Apply IIWT for LL, LH and HL subbands of R-Plane and modified HH band and create R-plane of the stego image.
- Step 13: Repeat Step 4 to Step12 for G&B planes by using black and white binary data image2&3.
- Step 14: Combine the all these layers into single stego image.
- Step 15: Transmit the created stego image to display screen.
- Step 16: Transmit the generated key secretly.



**Figure 4.** Eight patterns (a) Straight Up (SU), (b) Straight Down (SD), (c) Straight Forward (SF), (d) Straight Backward (SB), (e) Flipped Down (FD), (f) Flipped Up (FU), (g) Flipped Forward (FF), (h) Flipped Backward (FB).

The obtained stego images are given to the display screen inside the object showcase room. The flowchart representation of the proposed embedding algorithm is given in Figure 5.

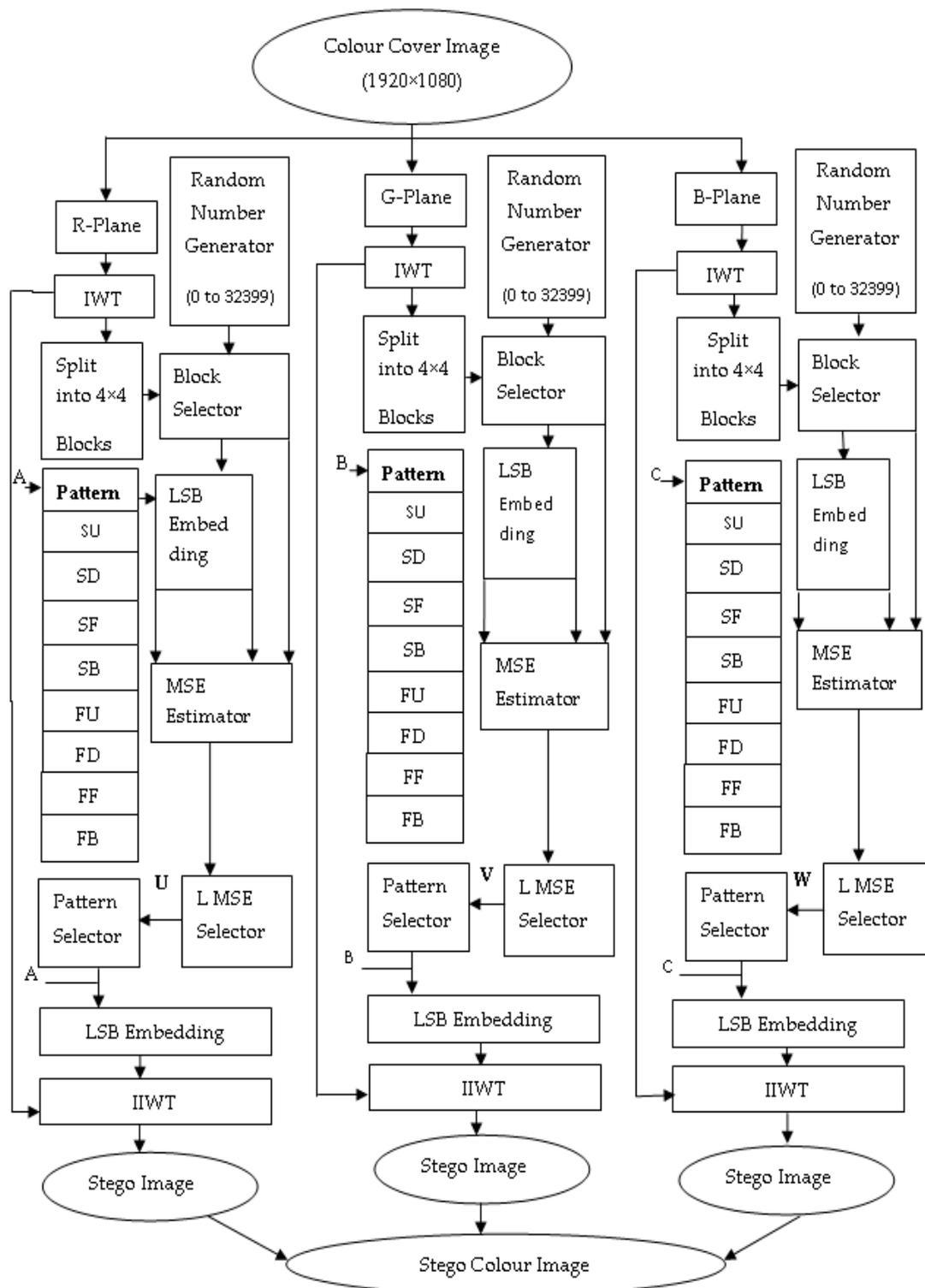


Figure 5. Flowchart representation of the embedding algorithm.

### 2.3. Key Generation and Reception

A dynamic key is generated for the proposed embedding algorithm based on the pattern that gives the least MSE for that block. A  $1920 \times 1080$  color image is divided into three planes. Each plane

HH subband is divided into  $4 \times 4$  blocks. Therefore, the total number of blocks for one plane is 32,400. For each block, the pattern with the least MSE is applied, and a 3-bit key for that pattern is obtained from Table 1. The key size for one plane is given as  $32,400 \times 3 = 97,200$  bits. Three different keys for three planes of the color image are generated using the steps given below.

- Step 1: Start the embedding algorithm for Red plane
- Step 2: Select the pattern with the least MSE for the selected block. Let this be U.
- Step 3: Look for the 3-bit key from Table 1 based on U.
- Step 4: Shift the selected key to left by three times.
- Step 5: Repeat Step 2 and Step 3 for the next selected block.
- Step 6: Replace the three LSBs of the key with the selected 3-bit key from Step 3.
- Step 7: Repeat Step 4.
- Step 8: Repeat from Step 5 to Step 7 until the key is generated for every  $4 \times 4$  block of the red plane.
- Step 9: Repeat from Step 1 to Step 8 for green and blue planes based on V and W respectively and generate keys for the green plane and blue plane.

The schematic representation of the key generation for a color image is given in Figure 6a–c. The keys are generated at the control center and are communicated to the object showcase room. There, the key bit stream is split into three bits and given to the proposed extraction algorithm.

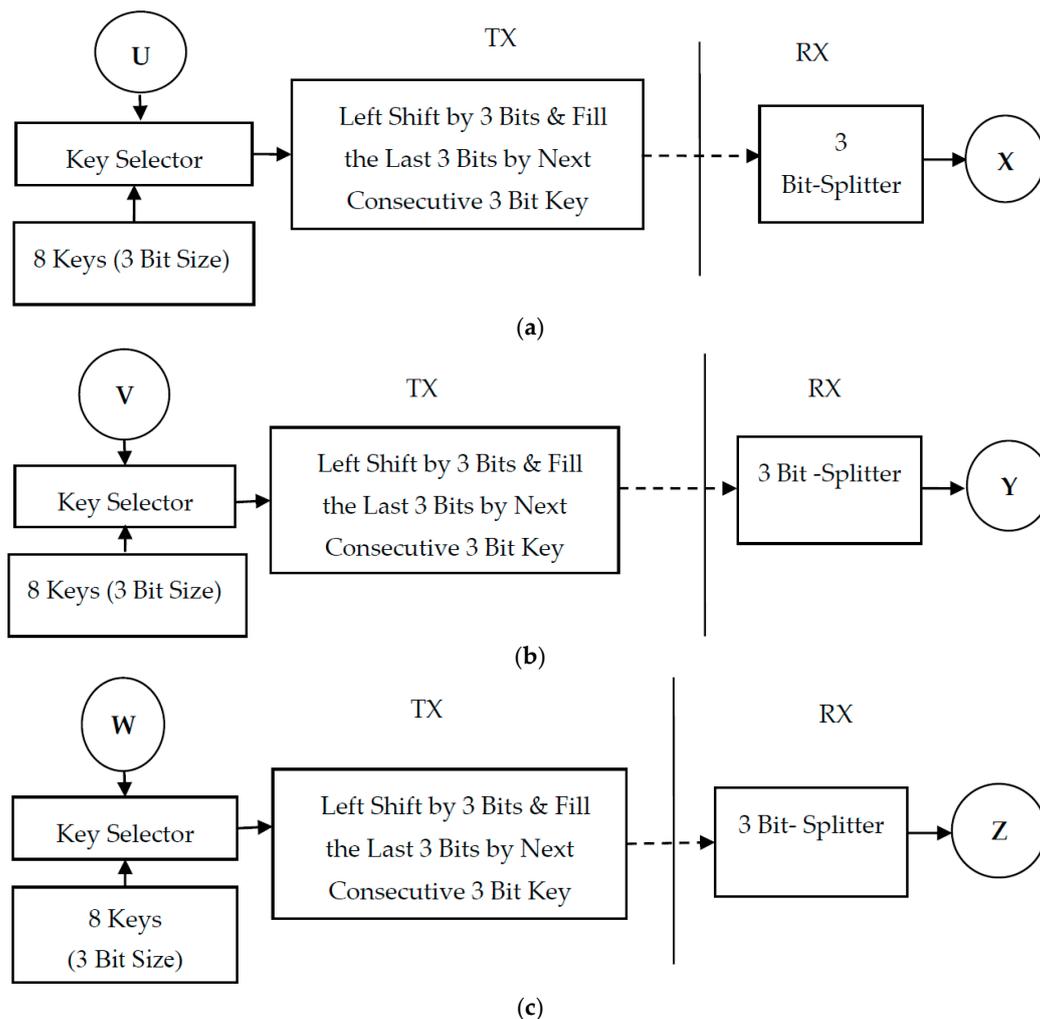


Figure 6. Key generation and reception (a) Red plane, (b) Blue Plane, (c) Green Plane.

**Table 1.** Three bit key for 8 patterns.

Pattern	Key
SU	000
SD	001
SF	010
SB	011
FU	100
FD	101
FF	110
FB	111

#### 2.4. Display Screen Extraction

The stego images created at the control room are fed to a display screen inside the object showcase room. The surveillance camera present inside the object showcase room captures the display screen along with the precious object and its surroundings. Now, the captured image is given to the object protection system, which is located inside the object showcase room, for message extraction. The process of extracting the display screen from the captured image is done at the sub-center. Various steps involving the display extraction from the captured image are given below.

Step 1: Store the captured image containing a display screen with minimum geometric distortion as a reference image in the grayscale form.

Step 2: Capture the image containing the display screen and convert into its grayscale form.

Step 3: Apply Harris corner detection for reference and captured images. This step gives the corner features of both the images.

Step 4: Extract the detected corner features of both images into two variables F1 and F2.

Step 5: Match the features between the two images using the variables F1 and F2. Let the matched features of the two images be M1 and M2.

Step 6: Find the valid matched points from the set of M1 and M2. Let the valid points be V1 and V2.

Step 7: Estimate the geometric transform required to restore the captured image as a reference image using V1 and V2.

Step 8: Apply the geometric transform to the captured color image.

Step 9: Perform edge detection on the restored image and find all the edges in the image.

Step 10: Apply Hough transform based on the results of edge detection. This gives Hough transform matrix H, line distance  $\rho$  and line inclination  $\theta$ .

Step 11: Find Hough peaks in the obtained Hough transform.

Step 12: Select lines having minimum line distance (borders of the display screen) inclining  $0 \pm 5^\circ$  (vertical lines) or  $90 \pm 5^\circ$  (horizontal lines).

Step 13: Lines obtained from Step 12 are used to extract the display screen from the captured image.

The extracted display screen is considered as the received stego image. This stego image is given to the proposed extraction algorithm, and the embedded data is extracted from the stego images. A flowchart representation of the display extraction process is given in Figure 7.

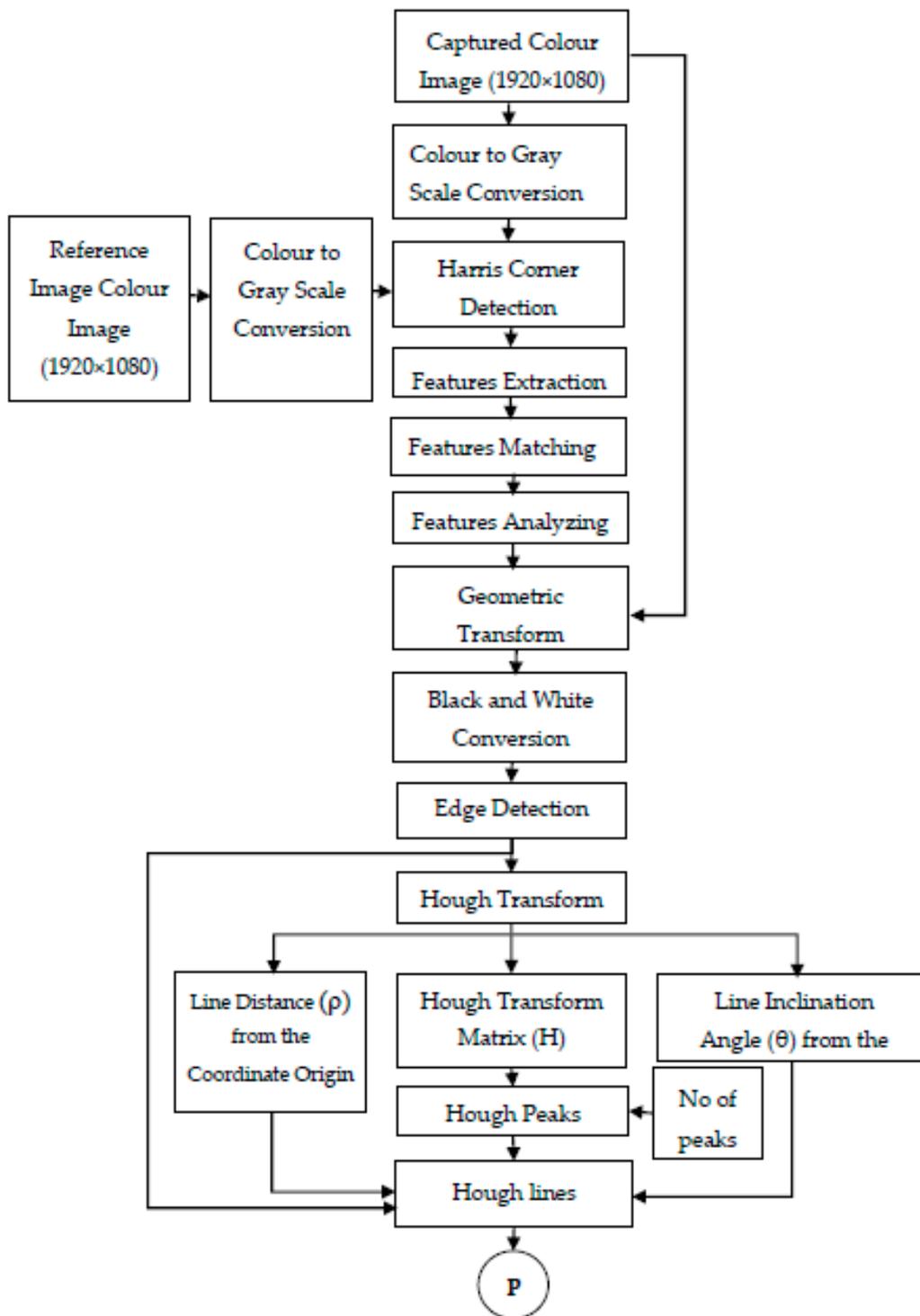


Figure 7. Flowchart representation for display screen extraction.

### 2.5. Extraction Algorithm

A flowchart representation of the proposed extraction algorithm is shown in Figure 8.

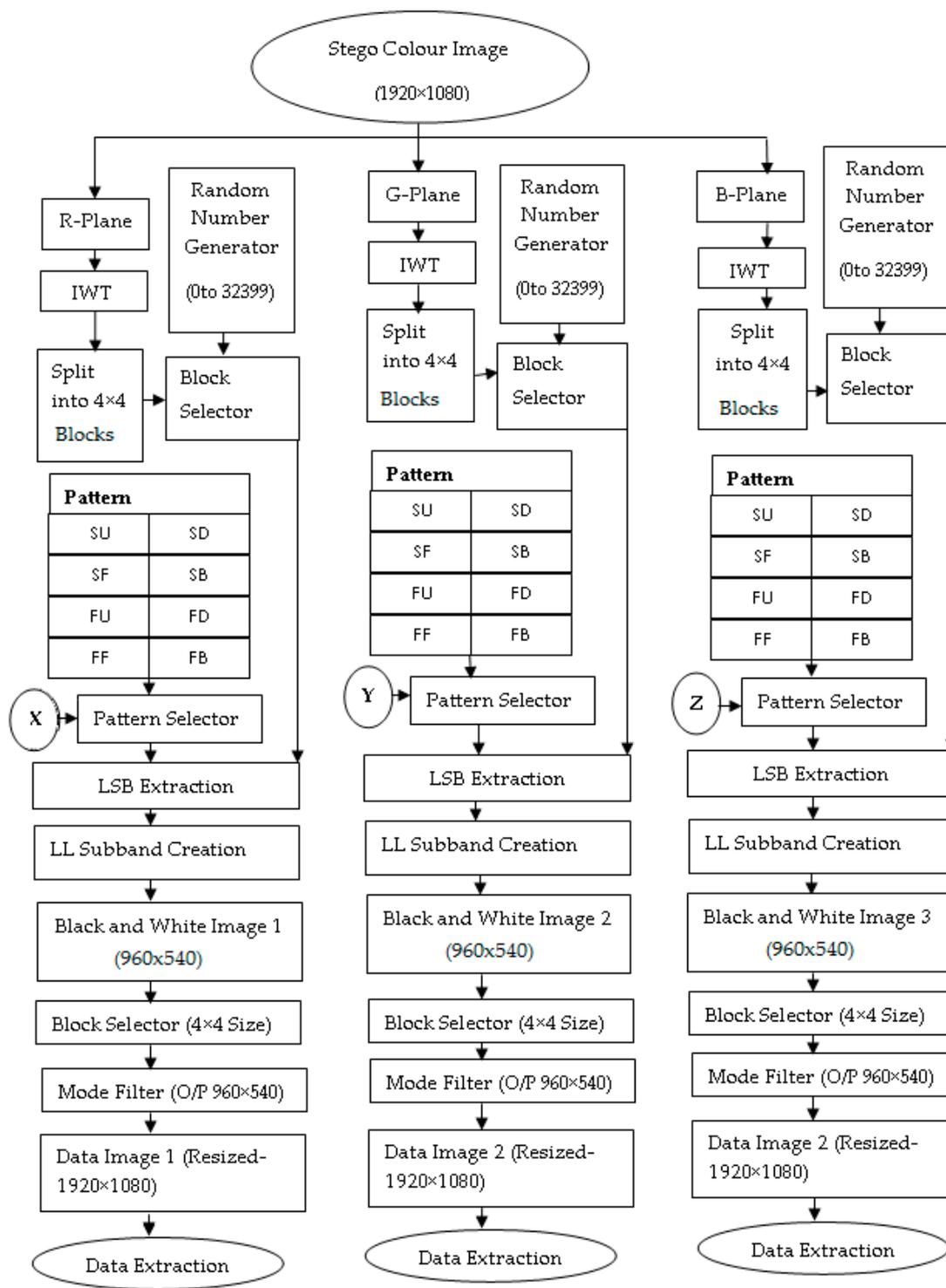


Figure 8. Flowchart representation of data extraction algorithm.

The stego images obtained from the display screen extraction process are re-sized to fit the cover image, i.e.,  $1920 \times 1080$ . The key bit stream transmitted by the control room is received and given to a 3-bit splitter. The extraction algorithm extracts the data image with inputs as stego images and keys. The data extraction process steps are given below.

- Step 1: Read the  $1920 \times 1080$  stego image obtained from display screen extraction.
- Step 2: Divide the stego image into its three layers red, green and blue.

- Step 3: Apply 2D-Haar IWT to R-Plane of the stego image and generate LL, LH, HL and HH band.
- Step 4: Select the HH band.
- Step 5: Divide HH band into  $4 \times 4$  blocks and select one block at a time randomly using same random number generator used at the transmitter.
- Step 6: The key stream received for the red layer is given to 3-bit splitter, and consecutive 3 LSB bits of the key are selected.
- Step 7: Shift the key to right by 3-bits.
- Step 8: The three bits of the key obtained from Step 6 is used to determine the pattern applied to that block by looking at Table 1.
- Step 9: Extract LSB of integer coefficients based on the selected pattern and create a  $4 \times 4$  block in the LL band of black and white binary data image respectively.
- Step 10: Select the next block and repeat from Step 6 to Step 9 until all data blocks of LL band are extracted from the HH band of the R-plane
- Step 11: Obtained LL subband gives black and white binary data image  $1(960 \times 540)$ .
- Step 12: Divide the obtained black and white data image into  $4 \times 4$  blocks.
- Step 13: Select the first block and apply mode filter to that block. Mode filter finds the most number of occurrences of '1' and '0' and replaces all the pixels in that block with '1' or '0', whichever is most prevalent.
- Step 14: Select next block in the same row and repeat Step 13 for that block. If the previous block is the last block in that row then, the next block is the first block from the next row.
- Step 15: Repeat Step 14 for all  $4 \times 4$  blocks and create restored black and white binary data image ( $960 \times 540$ ).
- Step 16: Resize the restored black and white data image ( $960 \times 540$ ) into the size of ( $1920 \times 1080$ ).
- Step 17: Repeat from Step 3 to Step 16 for green and blue layers using respective received key streams and create resized black and white binary data images.
- Step 18: Extract a secret message from the obtained black and white binary data image.

### 2.6. Object Detection and Tracking

The captured image from the surveillance camera is fed to the control room. The control room performs detection and tracking of the precious object inside the object showcase room. If the object is not present in the captured image frames, then a control signal is given to the museum alarm systems and the object protection system. The process of detecting and tracking the precious object is given below, and the flowchart representation of object detection and tracking is presented in Figure 9.

- Step 1: Read the initial or beginning image frame received from the surveillance camera.
- Step 2: Read the next or current image frame received from the surveillance camera.
- Step 3: Perform image segmentation for the obtained black & white image and select only the region that does not contain a display screen or region that contains the precious object.
- Step 4: Find the absolute difference between an initial frame and the current frame. This gives the distortions or disturbances between the two image frames. i.e., the movement of the precious object. This step results in a black and white image with the disturbances represented by white pixels and stationary locations represented by black pixels.
- Step 5: The white pixels in the black & white image represent movement of the precious object. Compute the centroid of all the white pixels in that image, i.e., the centroid of the precious object.
- Step 6: If there are no white pixels in the black & white image. Then, it is not possible to determine the centroid of the object.
- Step 7: Alert the museum alarm systems and send command information to the object protection system through the current stego image.

Step 8: Initialize the Kalman filter parameters [27] such as the covariance matrix, error in measurement matrix, error in estimation matrix, initial object location (centroid), and so on.

Step 9: Give the measured centroid value of the object to the Kalman filter; the output obtained is the estimated centroid value of the object.

Step 10: Assign the current image frame as an initial image frame for the next iteration.

Step 11: Repeat from Step 2 to Step 10 for continuous tracking of the precious object.

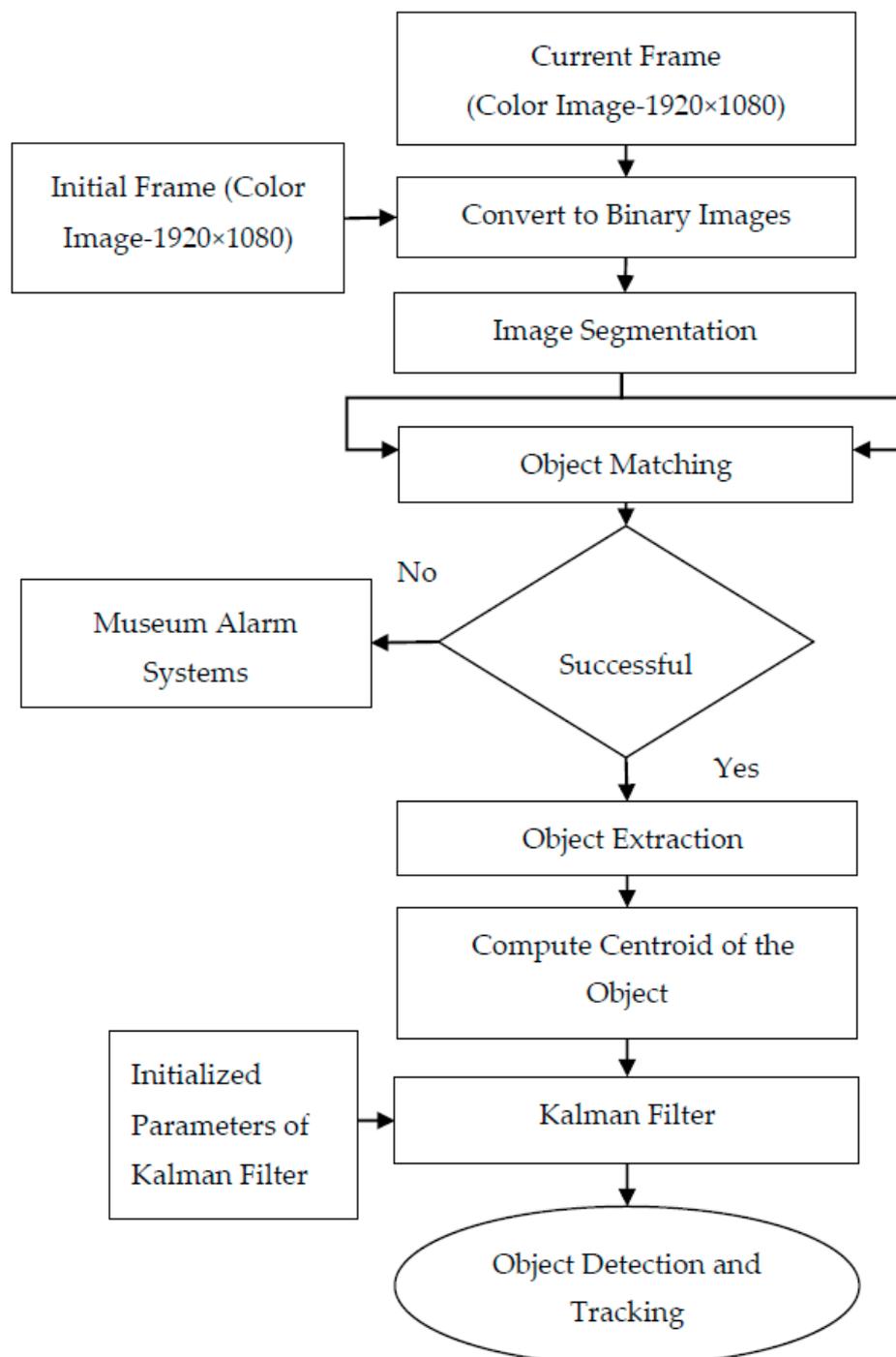


Figure 9. Flowchart representation for object detection and tracking.

### 3. Results and Discussion

The proposed system is designed using a visual display with  $1920 \times 1080$  pixels. Four color images of size  $1920 \times 1080$  are embedded with black and white data images, and stego images are created. These stego images are displayed on a display at a rate of 30 frames per second. These operations are performed using a PC with i5 INTEL core processor with 64-bit windows operating system. The clock speed of the system is 2.2 GHz and RAM is 8 GB. The four cover images are shown in Figure 10a–d. The black and white data images embedded into a red plane of these cover images are shown in Figure 11a–d, and their respective stego images are shown in Figure 12a–d. It is observed from the stego images that the embedded information is imperceptible to the human eye. The same process is repeated for green and blue planes respectively.

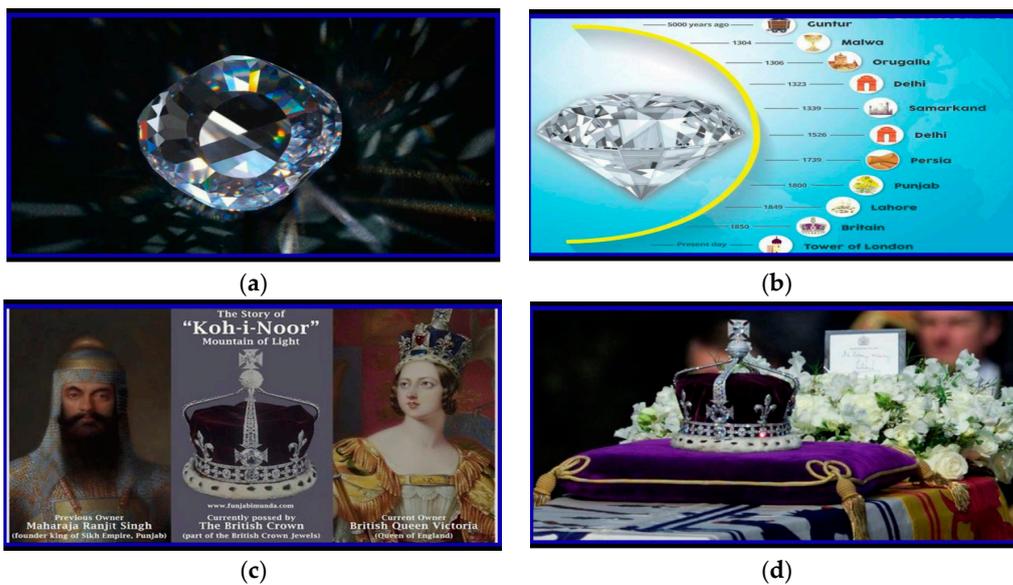


Figure 10. Cover Images (a) Image 1, (b) Image 2, (c) Image 3, (d) Image 4.

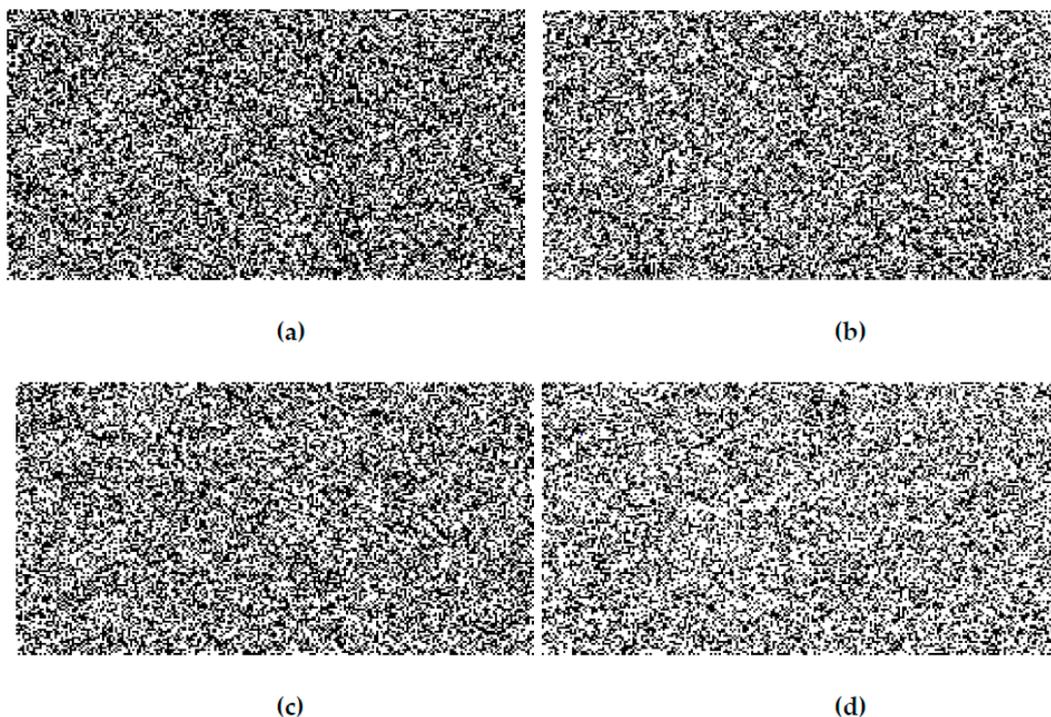


Figure 11. Black & White Data Images (a) Image 1, (b) Image 2, (c) Image 3, (d) Image 4.

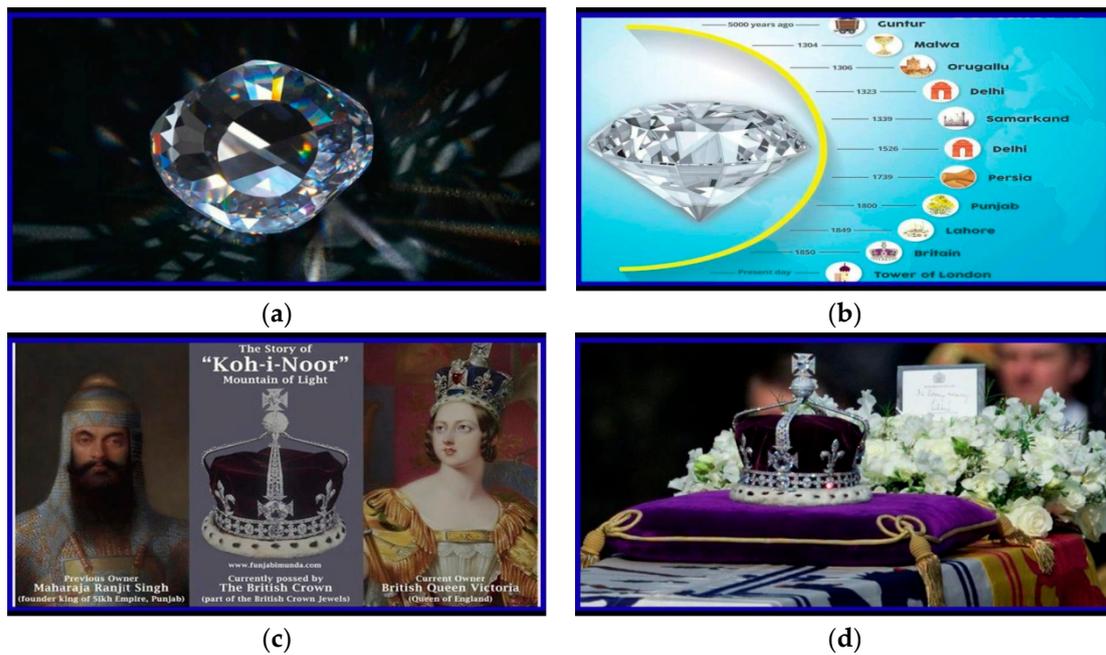


Figure 12. Stego Images (a) Image 1, (b) Image 2, (c) Image 3, (d) Image 4.

The image quality determines the performance of the proposed embedding algorithm. Image quality is the difference between the stego image and original image. The various statistical parameters used to evaluate the image quality are given below.

**Average Difference:** The Average Difference (AD) is the average of the difference between the cover image and the stego image, and is given by Equation (1). For the two same images, the average difference is zero. An average difference closer to zero indicates less distortion from the cover image.

$$AD = \frac{\sum_{i=1}^M \sum_{j=1}^N (O(i,j) - S(i,j))}{M \times N} \tag{1}$$

**Average Absolute Difference:** Average absolute difference (AAD) is the average of the absolute value of the difference between the cover image and the stego image, and is given by Equation (2). For the two same images, the average absolute difference is zero. An average absolute difference closer to zero indicates less distortion from the cover image.

$$AAD = \frac{\sum_{i=1}^M \sum_{j=1}^N (|O(i,j) - S(i,j)|)}{M \times N} \tag{2}$$

**Image Fidelity:** Image Fidelity (IF) is given Equation (3). For the two same images, the image fidelity is one. The image fidelity closer to one indicates less distortion from the cover image.

$$IF = 1 - \frac{\sum_{i=1}^M \sum_{j=1}^N (O(i,j) - S(i,j))^2}{\sum_{i=1}^M \sum_{j=1}^N (O(i,j))^2} \tag{3}$$

**Mean Square Error:** Mean Square Error (MSE) is given by Equation (4). For the two same images, the mean square error is zero. A mean square error closer to zero indicates less distortion from the cover image.

$$MSE = \frac{\sum_{i=1}^M \sum_{j=1}^N (O(i,j) - S(i,j))^2}{M \times N} \quad (4)$$

**Root Mean Square Error:** Root Mean Square Error (RMSE) is the square root of the mean square error, and is given Equation (5). For the two same images, the RMSE is zero. An RMSE closer to zero indicates less distortion from the cover image.

$$RMSE = \sqrt{\frac{\sum_{i=1}^M \sum_{j=1}^N (O(i,j) - S(i,j))^2}{M \times N}} \quad (5)$$

**Peak Signal to Noise Ratio:** Peak Signal to Noise Ratio (PSNR) is evaluated in decibels and is given by Equation (6). The higher value of PSNR indicates less distortion from the cover image. For a good quality image, PSNR is around 50 dB.

$$PSNR = 20 \log_{10} \left( \frac{O_{\max}}{\sqrt{MSE}} \right) \quad (6)$$

**Normalized Cross Correlation:** Normalized Cross Correlation (NK) is given by Equation (7). For the two same images, the NK is one. An NK closer to one indicates less distortion from the cover image.

$$NK = \frac{\sum_{i=1}^M \sum_{j=1}^N (O(i,j) \times S(i,j))}{\sum_{i=1}^M \sum_{j=1}^N (O(i,j))^2} \quad (7)$$

**Bit Error Rate:** Bit Error Rate (BER) is the measure of the number of error bits to the total number of bits and is given by Equation (8). BER varies from 0 to 1. For two same images, the BER is zero. A BER closer to zero indicates less distortion from the cover image.

$$BER = \frac{\text{Number of Erroneous Bits}}{\text{Total Number of Bits}} \quad (8)$$

**Structural Similarity Index Measurement:** Structural Similarity Index Measurement (SSIM) is given by Equation (9). For the two same images, the SSIM is one. An SSIM closer to one indicates less distortion from the cover image.

$$SSIM = \frac{(2 \times \bar{O} \times \bar{S} + C1) \times (2 \times \sigma_{OS} + C2)}{(\sigma_O^2 + \sigma_S^2 + C2) \times ((\bar{O})^2 + (\bar{S})^2 + C1)} \quad (9)$$

**Correlation:** Correlation (R) is given by Equation (10). For the two same images, the correlation is one. A correlation closer to one indicates less distortion from the cover image.

$$R = \frac{\left( \sum_{i=1}^M \sum_{j=1}^N (O(i,j) - \bar{O}) \times (S(i,j) - \bar{S}) \right)}{\sqrt{\left( \sum_{i=1}^M \sum_{j=1}^N (O(i,j) - \bar{O})^2 \right) \times \left( \sum_{i=1}^M \sum_{j=1}^N (S(i,j) - \bar{S})^2 \right)}} \tag{10}$$

**Structural Content:** Structural Content (SC) is given by Equation (11). For the two same images, the SC is one. An SC closer to one indicates less distortion from the cover image.

$$SC = \frac{\sum_{i=1}^M \sum_{j=1}^N (S(i,j))^2}{\sum_{i=1}^M \sum_{j=1}^N (O(i,j))^2} \tag{11}$$

These image quality parameters are evaluated for four cover images. The results of the proposed embedding algorithm are compared with the image quality parameters obtained from various familiar transform domain embedding techniques [25], such as Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), and Integer Wavelet Transform (IWT). The result for the single cover image (cover image1) is tabulated in Table 2. It is observed that the proposed algorithm produces better results when compared to existing algorithms.

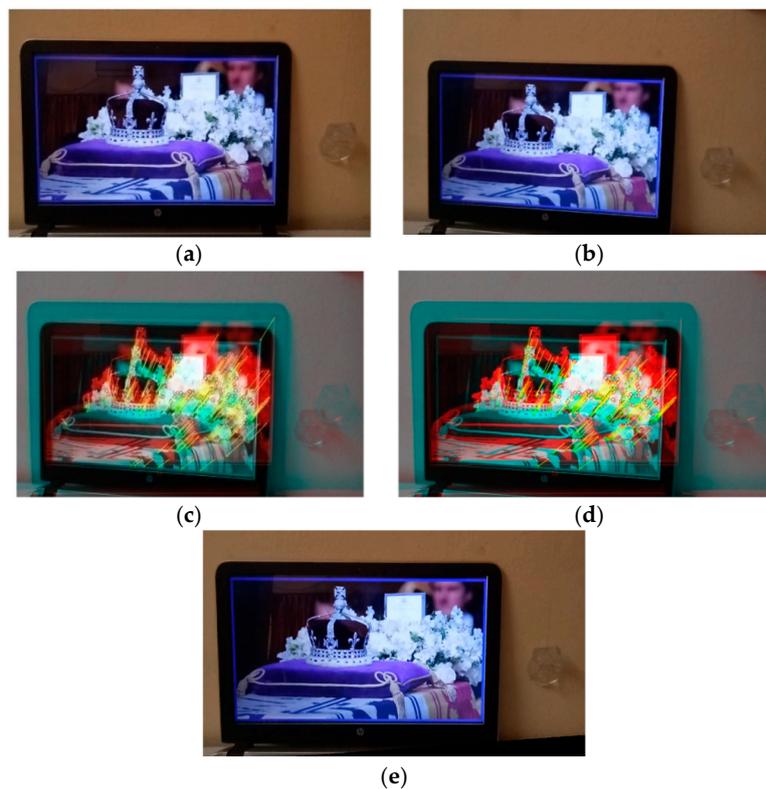
**Table 2.** Image analyses for cover image 1.

Parameter	IWT/ARC	IWT	DWT	DCT	DFT
AD	0.0298	0.0312	0.0487	0.0692	0.0724
AAD	0.6843	0.7574	0.7728	0.7822	0.7218
IF	0.9942	0.9870	0.9870	0.9868	0.9765
MSE	0.4815	0.6155	0.6602	0.6596	0.7432
RMSE	0.6939	0.7845	0.8125	0.8122	0.8621
PSNR	51.3048	50.238	49.934	49.938	48.775
NK	0.9985	0.9971	0.9973	0.9969	0.9965
BER	0.0198	0.0287	0.0288	0.0288	0.0315
SSIM	0.9886	0.9816	0.9714	0.9715	0.9843
R	0.9997	0.9989	0.9989	0.9989	0.9989
SC	0.9915	0.9868	0.9860	0.9859	0.9824

The stego images are displayed on the display screen and captured using a camera. The specifications of the camera are given in Table 3. A captured image with much less geometric distortion is considered as a reference image, as shown in Figure 13a. The captured image with geometric distortions is shown in Figure 13b. The Harris features are extracted from the both reference and captured images. The extracted features are matched for similarity, and the matching results are shown in Figure 13c. Out of all the matched features, only the features corresponding to the display screen are considered as valid features (Ref: Figure 13d). The geometric transform is estimated based on the valid features and applied for the captured image for restoration. The restored image is shown in Figure 13e.

**Table 3.** Camera Specifications.

Specification	Value
Resolution	13 Mega Pixels
Frame Rate	30 Frames per second
Width of the captured image	4160 pixels
Height of the captured image	3120 pixels



**Figure 13.** (a) Reference Image, (b) Captured Image, (c) Matched Features, (d) Valid Features, (e) Restored Image.

A Hough transform is applied to the restored image and lines are detected in the restored image, as shown in Figure 14. Only vertical and horizontal lines having line distance equivalent to the border of the display screen are selected by restricting the theta values to  $0 \pm 5^\circ$  and  $90 \pm 5^\circ$  respectively. The selected lines are used to extract the received stego image, as shown in Figure 15a–d. The Four received stego images are given to the extraction algorithm along with the received keys, and black & white binary data images are extracted from the red plane, as shown in Figure 16a–d. The received black & white images contain noise due to the channel. These noises are filtered out using the mode filter, and the black & white binary data images are restored, as shown in Figure 17a–d. The restored image of size  $(960 \times 540)$  is resized into  $(1920 \times 1080)$ , as shown in Figure 18a–d, to achieve the required data capacity. This process is repeated for green and blue planes respectively.



**Figure 14.** Hough Line Detection, (\*blue color—actual line present in the cover image, pink color—detected Hough lines, and green color—detected ROI).

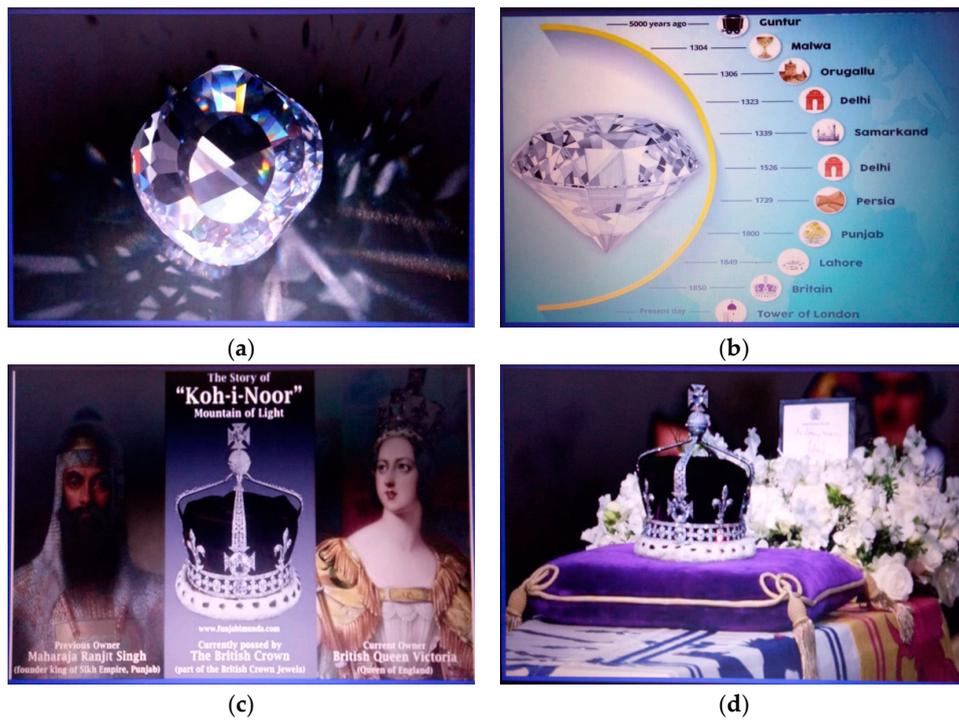


Figure 15. Received Stego Image(a) Image 1, (b) Image 2, (c) Image 3, (d) Image 4.

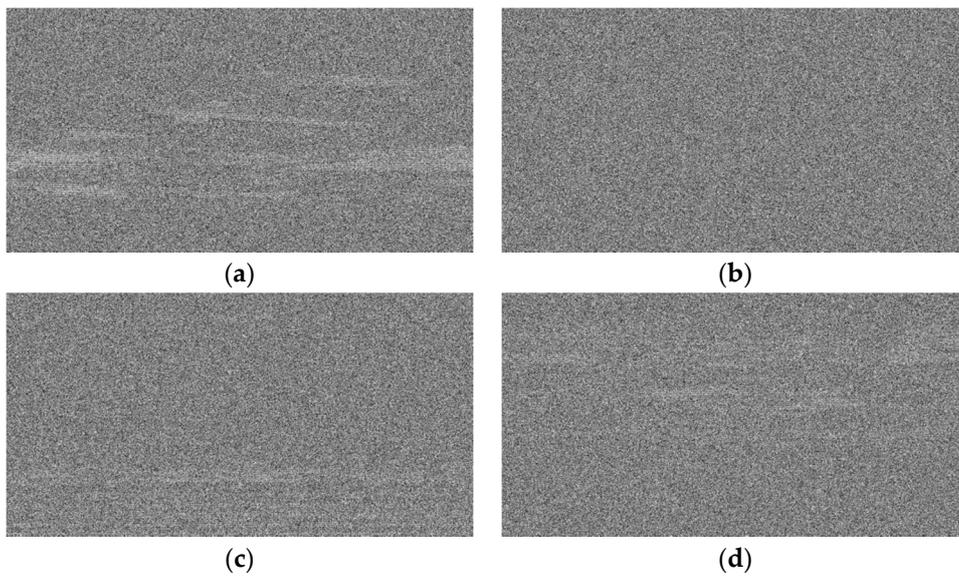


Figure 16. Extracted data image (a) Image 1, (b) Image 2, (c) Image 3, (d) Image 4.

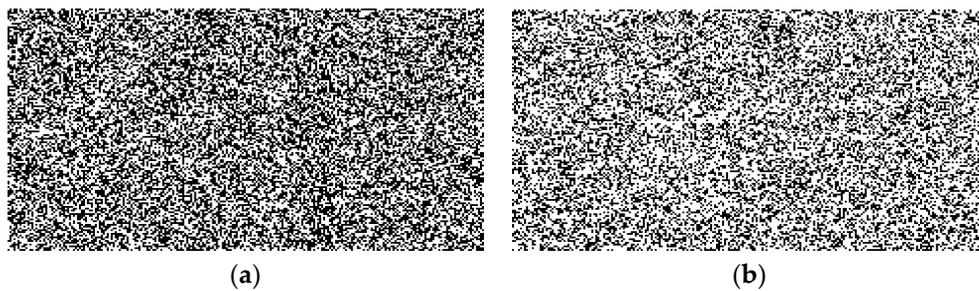


Figure 17. Cont.

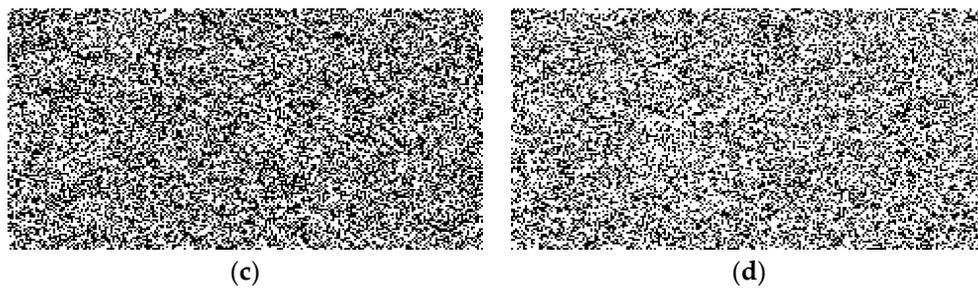


Figure 17. Restored Black & White Data Image (a) Image 1, (b) Image2, (c) Image 3, (d) Image 4.

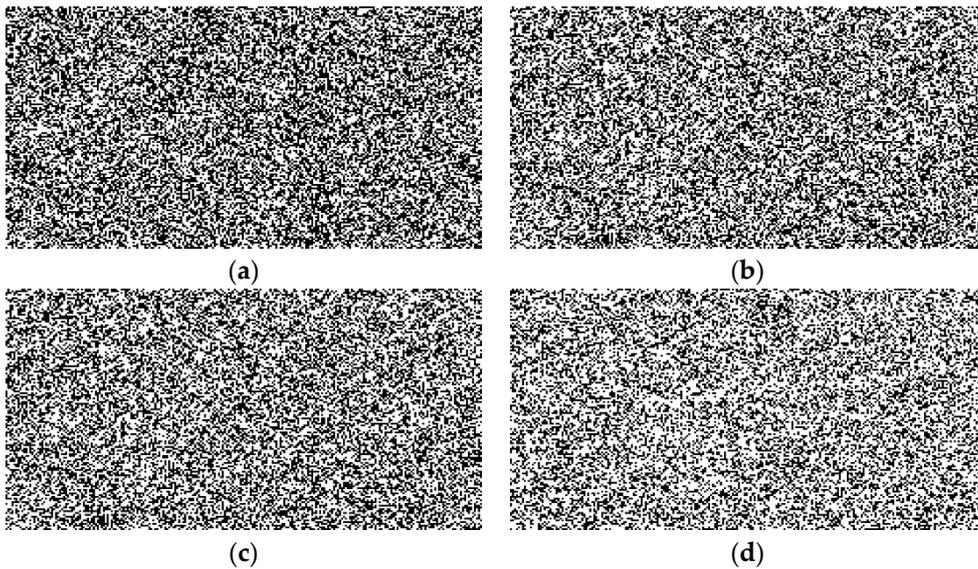


Figure 18. Resized Black& White Data Image (a) Image 1, (b) Image 2, (c) Image 3, and (d) Image 4.

**Complexity Estimation:**

In the proposed algorithm, the  $960 \times 540$  HH subband of the cover image is divided into  $4 \times 4$  blocks. Therefore, the total number of blocks is 32,400. Each block is selected randomly until every LL block of the data image is embedded within HH band of the cover image. The selection blocks can be done in  $32,400!$  ways. In each block, eight different patterns are applied, and the pattern with the least MSE is selected for that block. This can be done in 8 ways. Therefore, for a brute force attack, it takes  $32,400! \times 8$  ways to break the security of the proposed algorithm.

**Accuracy and Bit Rate:**

At the receiver, the accuracy of the proposed system is estimated for all four images using both the proposed and pre-existing embedding algorithms. The results are shown in Table 4. It was observed that the proposed embedding algorithm produces an accuracy of 97.6%, which is significantly higher than that of existing algorithms.

$$\text{Accuracy}(\%) = \left( \frac{\text{Total Number of Blocks} - \text{Number of Erroneous Blocks}}{\text{Total Number of Blocks}} \right) \times 100 \quad (12)$$

$$\text{Bit Rate} = (\text{Number of Blocks} \times \text{Number of Frame} \times \text{Number of Planes}) \quad (13)$$

**Table 4.** Accuracy comparisons of the proposed system for various algorithms.

Algorithm	Cover Image 1	Cover Image 2	Cover Image 3	Cover Image 4
DFT	94.6	95.2	94.9	95.3
DCT	93.4	93.2	93.6	94.0
DWT	95.8	96.8	95.6	95.7
IWT	96.2	97.1	96.6	97.2
IWT/ARC	96.8	97.3	97.6	97.6

Furthermore, the accuracy and bit rate for different  $n \times n$  blocks are compared for  $1920 \times 1080$  images, as shown in Table 5 using Equations (12) and (13). It was observed that for  $n=16$  the accuracy was high but the bit rate was low, and for  $n = 4$ , the accuracy was low but the bit rate was high. Therefore, it is suggested that  $n = 8$  be used, which provides good accuracy and bit rate.

**Table 5.** Accuracy and Bit Rate comparison.

Block Size	$4 \times 4$	$8 \times 8$	$16 \times 16$
Accuracy (%)	91.1	97.6	98.2
Bit Rate (bits/s)	11,664,000	2,916,000	729,000

The time complexity of the proposed algorithm is estimated for four cover images in the following scenario. Three pieces of command information, namely Normal ('NO'), Alert1 ('A1'), Alert2 ('A2') normally used in museum security services, are considered. These commands are hidden over one plane of the four cover images and extracted at the receiver. The block size has been increased to embed these commands of size 16 bit. The time complexity of the proposed algorithm is analyzed in the given scenario and illustrated in Table 6. It was observed that the proposed algorithm was capable of transmitting commands in less than 0.1 s, which is sufficient for museum security applications.

**Table 6.** Time complexity analysis.

Cover Images	Time Complexity in ms			
	Command1 = 'NO'	Command2 = 'A1'	Command3 = 'A2'	Average
Image 1	94.7769	93.8743	95.7877	94.8130
Image 2	91.0290	91.1963	90.7975	91.0076
Image 3	96.1845	94.0191	96.7095	95.6377
Image 4	94.5715	91.6055	93.0815	93.0862
	<b>Mean</b>			<b>93.6361</b>

At the control room, in each received frame, the region-II portion of the image shown in Figure 19 is segmented and is given to the Kalman filter for object detection and tracking. The tracking results for the six frames are given in Figure 20a–f. The green circle represents the Kalman filter estimation of the object, and the black circle represents the true position of the object. The graphical representation of the true and estimated positions movement of the precious object is given in Figure 21a,b. If the Kalman filter fails to estimate the centroid of the precious object, then an alert signal is given to the museum alarm and computer system.

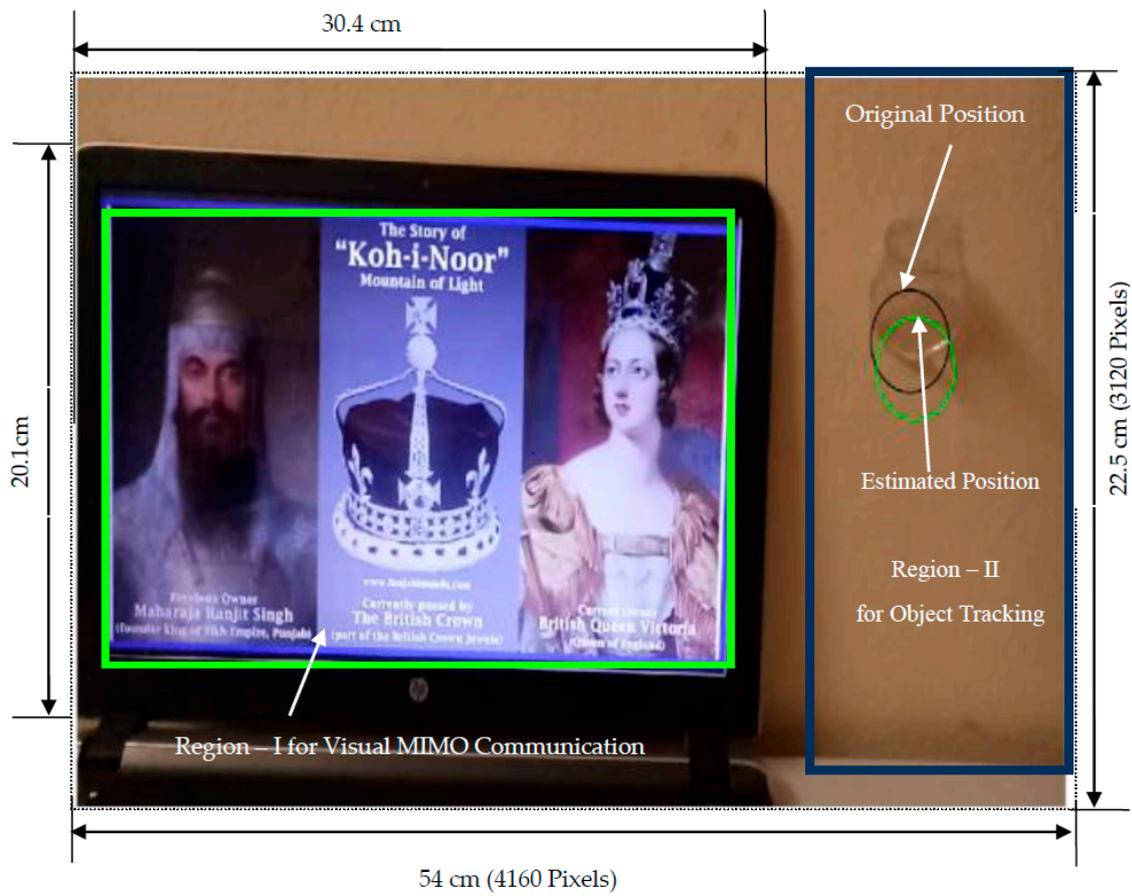


Figure 19. ROI for object tracking.

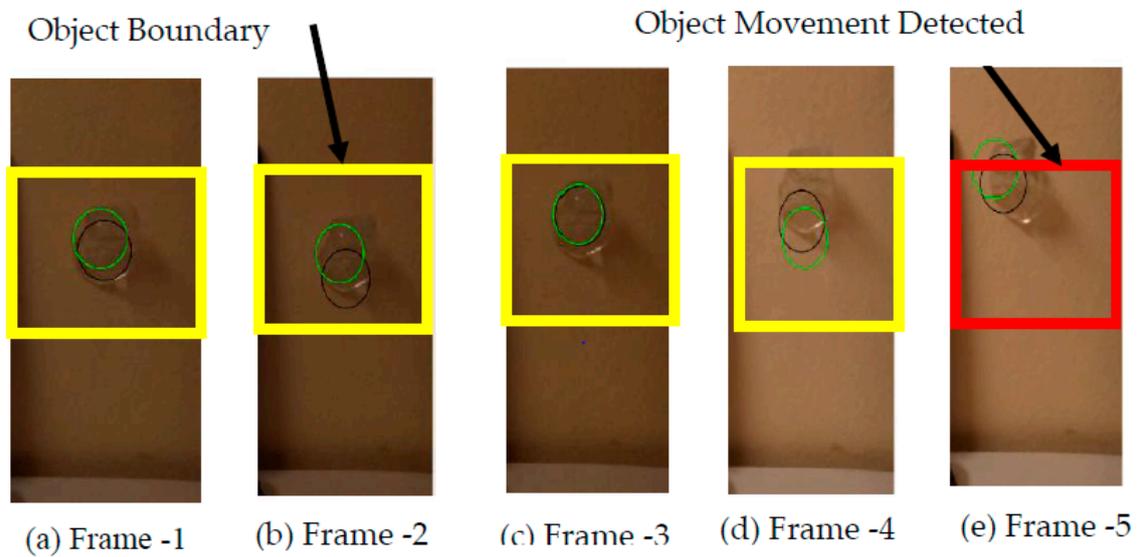
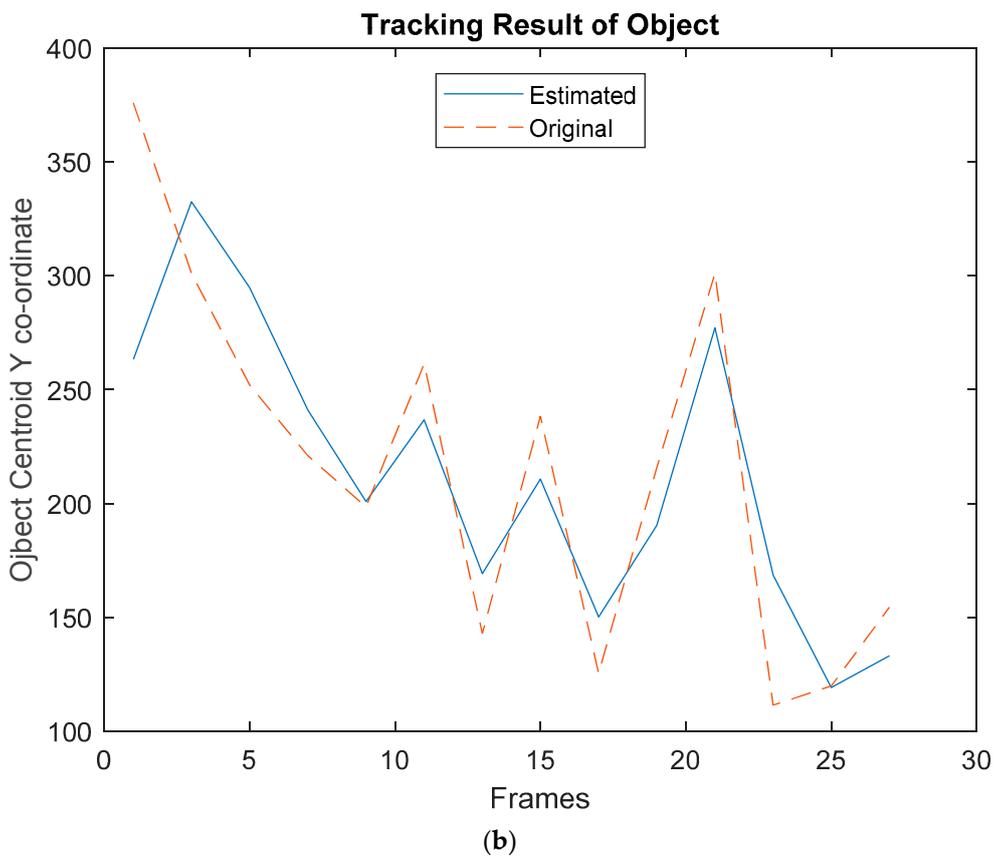
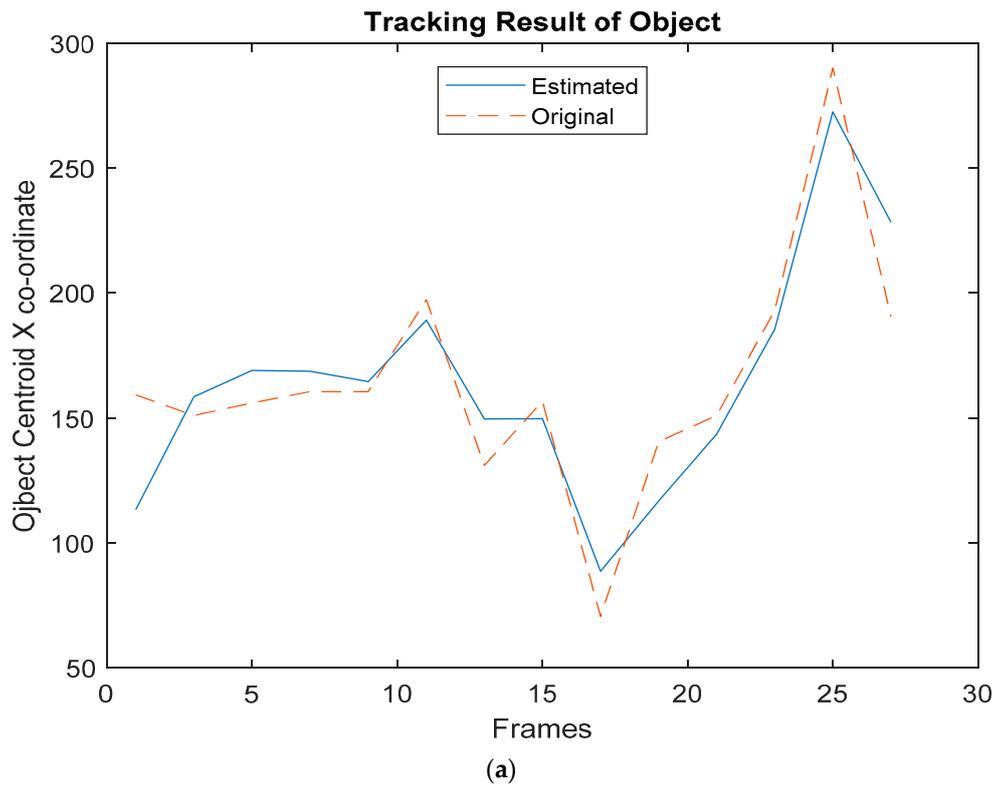


Figure 20. Tracking result of six frames.



**Figure 21.** Tracking result of object centroid X and Y coordinates (a) X co-ordinate Tracking (b) Y co-ordinate Tracking.

The performance of the proposed object tracking algorithm is evaluated in terms of its accuracy in the XY-plane. The actual center position of the object ( $x_{act}, y_{act}$ ) is observed visually and noted in terms of cm. The position of the object ( $x_{est}, y_{est}$ ) estimated from the image will be in terms of pixels. Hence, a resolution factor ( $X_{res}, Y_{res}$ ) correlating the physical dimensions to the pixels of the images is evaluated, as in Equations (14) and (15), for X and Y-axis respectively. It can be used to bring down the estimated and actual position of the object to a common domain, which is required to evaluate the error metrics. To normalize the error metrics, the tracking error is evaluated per unit of object dimension A round object of 5 cm diameter is tracked in the proposed work, and the mean tracking error over 'N' frames are evaluated as in Equation (16) for X-axis and Equation (17) for Y-axis. Finally, the tracking accuracies of X and Y plane are determined using Equations (18) and (19) respectively. The experimental results illustrate a significant tracking accuracy of 95.54% and 98.54% in the X and Y coordinates respectively. This demonstrates the performance of the proposed work in terms of providing a reliable tracking mechanism for precious objects, and enhances museum security.

$$X_{res} = \frac{\text{Height of the captured image in pixels}}{\text{Physical height of the image in cm}} \quad (14)$$

$$Y_{res} = \frac{\text{Width of the captured image in pixels}}{\text{Physical width of the image in cm}} \quad (15)$$

$$E_x(\%) = \left( \frac{1}{N} \sum_{i=1}^N \left| \left( x(i)_{act} - \frac{x(i)_{est}}{X_{res}} \right) \right| \div D \right) \times 100 \quad (16)$$

$$E_y(\%) = \left( \frac{1}{N} \sum_{i=1}^N \left| \left( y(i)_{act} - \frac{y(i)_{est}}{Y_{res}} \right) \right| \div D \right) \times 100 \quad (17)$$

$$A_x(\%) = 100 - E_x \quad (18)$$

$$A_y(\%) = 100 - E_y \quad (19)$$

#### 4. Conclusions

The proposed system makes use of existing display screen and surveillance cameras facilities. This reduces the overhead cost and need for additional components. It provides a robust, secure MIMO communication services without distorting the quality of the cover image displayed on the screen. Spatial coverage quality of a single camera enables invisible simultaneous visual MIMO communication and object monitoring services. Hough transform is used to extract ROI of stego image, and object tracking is achieved by a Kalman filter-based template matching technique. An IWT-based ARC-LSB embedding technique has been adopted for image hiding and extraction. Color independent visual MIMO is employed, which provides a high data rate. A combination of  $32,400! \times 8$  is required to recover the data image, which illustrates the robustness of the proposed embedding algorithm.

Performance of the proposed system is analyzed in terms of imperceptibility and accuracy at the transmitter and receiver terminals. Experiments results at the transmitter illustrate a significant improvement in imperceptibility as compared with existing techniques. Similarly, at the reception terminal for the received data image, experimental results show a promising accuracy of 97.6%, while for object tracking, accuracy of 95% is observed.

Further work will include an investigation of the performance assessment of the proposed system in the presence of distortions in light illumination, relative distance, and orientation. Usage of randomization algorithms for data images created to provide additional security is also a future initiative associated with this present work. This work can also be extended to control violence against civilians by providing 24/7 hidden security services in all public gathering places by implementing suspicious-movement detection and multi-object tracking algorithms in the proposed system.

**Author Contributions:** All the authors equally participated in technical discussion, design, implementation, testing, performance measurement and writing the article.

**Funding:** This research received no external funding.

**Acknowledgments:** We would like to thank god for giving us an opportunity, encouragement, and strength to finish this work.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Hashim, A.F.; Taib, M.Z.F.; Alias, A. The Integration of Interactive Display Method and Heritage Exhibition at Museum. *Procedia Soc. Behav. Sci.* **2014**, *44*, 308–316. [[CrossRef](#)]
2. Tsiropoulou, E.E.; Thanou, A.; Paruchuri, S.T.; Papavassiliou, S. Modelling Museum Visitors' Quality of Experience. In Proceedings of the IEEE 11th International Workshop on Semantic and Social Media adaptation and Personalization (SMAP), Thessaloniki, Greece, 20–21 October 2016; pp. 1–6. [[CrossRef](#)]
3. Tsiropoulou, E.E.; Thanou, A.; Papavassiliou, S. Self-organizing museum visitor communities: A participatory action research based approach. In Proceedings of the IEEE 12th International Workshop on Semantic and Social Media Adaptation and Personalization (SMAP), Bratislava, Slovakia, 7–9 July 2017; pp. 1–5. [[CrossRef](#)]
4. Lykourantzou, L.; Claude, X.; Naudet, Y.; Tobias, E.; Antoniou, A.; Lepouras, G.; Vasilakis, C. Improving museum visitors' Quality of Experience through intelligent recommendations: A visiting style-based approach. In Proceedings of the Workshop of the 9th International Conference on Intelligent Environments, Athens, Greece, 16–17 July 2013; pp. 507–518. [[CrossRef](#)]
5. Tsiropoulou, E.E.; Thanou, A.; Papavassiliou, S. Quality of experience-based museum touring: A human in the loop approach. *Soc. Netw. Anal. Min.* **2017**, *7*. [[CrossRef](#)]
6. Fernández, J.; Calavia, L.; Baladrón, C.; Aguiar, J.M.; Carro, B.; Sanchez-Esguevillas, A.; Alonso-López, J.A.; Smilansky, Z. An intelligent surveillance platform for large metropolitan areas with dense sensor deployment. *Sensors* **2013**, *13*, 7414–7442. [[CrossRef](#)] [[PubMed](#)]
7. Sookhanaphibarn, K.; Thawonmas, R. A Movement Data Analysis and Synthesis Tool for Museum Visitors' Behaviors. In *PCM 2009: Advances in Multimedia Information Processing*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 144–154. [[CrossRef](#)]
8. Lin, S.; Chou, S. Museum Visitor Routing Problem with the Balancing of Concurrent Visitors. In *Complex Systems Concurrent Engineering*; Springer: London, UK, 2017; pp. 345–353. [[CrossRef](#)]
9. Saha, N.; Shareef Iftekhar, M.; Le, N.T.; Jang, Y.M. Survey on optical camera communications: Challenges and opportunities. *IET Optoelectron.* **2015**, *9*, 172–183. [[CrossRef](#)]
10. Ghassemlooy, Z.; Zvanovec, S.; Khalighi, M.A.; Popoola, W.O.; Perez, J. Optical wireless communication systems. *Optik* **2017**, *151*, 1–6. [[CrossRef](#)]
11. Ashok, A.; Gruteser, M.; Mandayam, N.; Silva, J.; Varga, M.; Dana, K. Challenge: Mobile Optical Networks through Visual MIMO. In Proceedings of the Sixteenth Annual International Conference on Mobile Computing and Networking, Chicago, IL, USA, 20–24 September 2010; pp. 105–112. [[CrossRef](#)]
12. Ashok, A.; Gruteser, M.; Mandayam, N.; Kwon, T.; Yuan, W.; Varga, M.; Dana, K. Rate Adaptation in Visual MIMO. In Proceedings of the 2011 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), Salt Lake City, UT, USA, 27–30 June 2011; pp. 583–591. [[CrossRef](#)]
13. Ashok, A.; Gruteser, M.; Mandayam, N.; Dana, K. Characterizing multiplexing and diversity in visual MIMO. In Proceedings of the IEEE 2011 45th Annual Conference on Information Sciences and Systems (CISS), Baltimore, MD, USA, 23–25 March 2011; pp. 1–6. [[CrossRef](#)]
14. Yuan, W.; Dana, K.; Varga, M.; Ashok, A.; Gruteser, M.; Mandayam, N. Computer vision methods for visual MIMO optical system. In Proceedings of the CVPR 2011 WORKSHOPS, Colorado Springs, CO, USA, 20–25 June 2011; pp. 37–43. [[CrossRef](#)]
15. Varga, M.; Ashok, A.; Gruteser, M.; Mandayam, N.; Yuan, W.; Dana, K. Visual MIMO based led-camera communication applied to automobile safety. In Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services, Bethesda, MD, USA, 28 June–1 July 2011; pp. 383–384. [[CrossRef](#)]

16. Kim, J.E.; Kim, J.W.; Kim, K.D. LEA detection and tracking method for color-independent visual-MIMO. *Sensors* **2016**, *16*, 1027. [[CrossRef](#)] [[PubMed](#)]
17. Manikanda, C.; Neelamegam, P.; Wilfred, A.A. Visual-MIMO for vehicle to vehicle communications. In Proceedings of the IEEE 2017th Microelectronic Devices, Circuits and Systems (ICMDCS), Vellore, India, 10–12 August 2017; pp. 1–6. [[CrossRef](#)]
18. Tan, J.J.; Zou, C.Q.; Du, S.H.; Tan, J.T. Simulation of MIMO channel characteristics for indoor visible light communication with LEDs. *Optik* **2014**, *125*, 44–49. [[CrossRef](#)]
19. Yuan, W.; Dana, K.J.; Ashok, A.; Gruteser, M.; Mandayam, N.B. Spatially varying radiometric calibration for camera-display messaging. In Proceedings of the 2013th IEEE Global Conference on Signal and Information Processing, Austin, TX, USA, 3–5 December 2013; pp. 763–766. [[CrossRef](#)]
20. Manikandan, C.; Neelamegam, P.; Rakesh Kumar, S.; Sai Siva Satwik, K. A novel visual SISO and MIMO computer Interface for process production system. *Int. J. Mech. Prod. Eng. Res. Dev.* **2018**, *8*, 181–188. [[CrossRef](#)]
21. Yuan, W.; Dana, K.; Ashok, A.; Gruteser, M.; Mandayam, N. Dynamic and invisible messaging for visual MIMO. In Proceedings of the IEEE Global Conference on Signal and Information Processing, Austin, TX, USA, 3–5 December 2013; pp. 346–352. [[CrossRef](#)]
22. Jambhekar, N.; Dhawale, C.; Hegadi, R. Performance Analysis of Digital Image Steganographic Algorithm. In Proceedings of the International Conference on Information and Communication Technology for Competitive Strategies, Rajasthan, India, 14–16 November 2014; pp. 1–7. [[CrossRef](#)]
23. Sidhik, S.; Sudheer, S.; Pillai, V.M. Performance and analysis of high capacity steganography of color images involving wavelet transform. *Optik* **2015**, *126*, 3755–3760. [[CrossRef](#)]
24. Sai Siva Satwik, K.; Manikandan, C.; Elamaran, V.; Narasimhan, K.; Raju, N. An adaptive row-column least significant bit inlay approach for image steganography. *Biomed. Res.* **2017**, *28*, 10216–10222.
25. Manikandan, C.; Sai Siva Satwik, K.; Rakesh Kumar, S.; Neelamegam, P.; Venkatesh, S. Performance Analysis of Visual MIMO Communication System with Various Transform Domain Embedding Algorithms. In Proceedings of the International Conference on Electrical Sciences, Osaka, Japan, 8–9 August 2018; pp. 1–6.
26. Liu, W.; Zhang, Z.; Li, S.; Tao, D. Road detection by using a generalized hough transform. *Remote Sens.* **2017**, *9*, 590. [[CrossRef](#)]
27. Jeong, J.M.; Yoon, T.S.; Park, J.B. Kalman filter based multiple objects detection-tracking algorithm robust to occlusion. In Proceedings of the SICE Annual Conference (SICE), Sapporo, Japan, 9–12 September 2014; pp. 941–946. [[CrossRef](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).