Interpretation of Spatial Relationships by Objects Tracking in a Complex Streaming Video

Noralhuda N. Alabid[1]

ABSTRACT: Interpretation of spatial relations among objects provides the cornerstone of image processing of several computing systems and applications, such as the processing and the running of video surveillance applications, the directing of robotics, and the development of scenes understanding systems. The models of spatial relationships have been well investigated in a two-dimensional scene, but due to the recent advancement in technology, the processing in a three-dimensional scene becomes applicable as well. However, to the best of the author's knowledge, interpretation of the spatial relations has been well explored between silent objects in images, but not in the case of the dynamic spatial relation between a moving object and another silent one. Thus, this study aims to explore this type of processing and in a time-varying scene rather than in an only silent scene. The spatial relationships were determined by using the motion-based object tracking model along with the Hypergraph-Object Oriented Model (HOOM). Defining the types of spatial relationships was conducted based on two strategies; determining each object with a bounding box and then comparing the boxes' locations by applying certain conditional rules. This study identified some of the spatial relationships in a three-dimension scene of streaming frames, which were "Direct in front of", "In front of on the right/left", "Direct behind of", "Behind of on the right/left", "To the right", "To the left", "On", "Under", "Besides", and "Besides on the right/left". In addition, conducting the relationships of moving object "Between" two silent objects was also carried out. The experimental results obtained based on actual indoor streaming frames show the effectiveness and reliable execution of the system

Keywords: Face Recognition, Features Extraction, Image Segmentation, Spatial Relations, Tracking Objects

DOI: 10.37936/ecti-cit.2021152.240953

Article history: received May 31, 2020; revised October 15, 2020; accepted November 1, 2020; available online May 5, 2021

1. INTRODUCTION

The growth in the field of optical technology and computer vision resulted in an urgent need for finding a robust system that can analyse images' contents accurately. Conducting the models of spatial relationships in digital media provides valuable information required for scientific and commercials purposes. For example, the process of information retrieval of spatial data has been used extensively in Geographical Information Systems (GIS) to provide location services that rely on visualizations and geographic analysis, the core component of spatial relations analysis in GIS is Machine Learning (ML) [1]. The models of spatial relationships are essentials in different areas of computer vision, the robotic industry relied on computer vision systems to provide the main information in robotic control systems [2].Here, the study aims to build a model exercised to analyse video's contents based on specific dynamic spatial relations between objects using a Hypergraph-Object Oriented Model (HOOM) [3], [4]. HOOM is applied to determine the geometric inference for the interest objects in the main scene. Among the approaches applied here were Object-Oriented Programming (OOP) used for image processing, image segmentation used for determining the position of entities, and extraction of the efficient content used for performing the geometric inference.

[1]The author is with Department of Computer Science, Faculty of Education, University of Kufa, Najaf, Iraq.,E-mail: noralhudan.hadi@uokufa.edu.iq

[3]Corresponding author: noralhudan.hadi@uokufa.edu.iq, ORCID ID: 0000-0001-5695-1532

The model relied on finding the coordinates of the object's boundaries boxes, particularly the boxes centre. By comparing the coordinates of the centre and the border of each object's box with each other, the type of spatial relationship was defined. This paper is organized as follows: section two discusses the issue related to object detection and tracking, machine learning algorithms, and the methods to determine the spatial relations between objects. Experiential results are presented in section four, while section five concludes the results.

2. RELATED WORKS

2.1 Machine learning algorithms

Recognition systems depend mainly on complex mathematical calculations and ML algorithms. Most applications of ML are focused on the theory, performance, and characteristics of learning systems and algorithms. ML is an interdisciplinary field of research implemented in a variety of computer fields of science such as artificial intelligence, statistics, cognitive science, etc. It has been used on a variety of problems in which the recognition system is one of them [5], [6]. ML implemented models are supplied with a set of algorithms supported discrimination processes among a variety of features, expressions, behaviours, and several other aspects. Therefore, a process of matching between for example captured images and those stored pre-excited in the database can be treated with aid of ML [6], [7]. ML approach to image recognition requires finding and extracting reliable key features from images and utilizing them as input data to a model of ML. The importance of ML is appeared from the inability of experts to choose suitable reliable features in an image that should have sufficient discrimination power to perform the tasks of recognition. Too many redundant features may result from the inability to extract reliable general local spatial structures of an image, due to the inability of human cognition to choose the appropriate features. However, ML can perform this challenge mission, recognize the reliable features can be gained by training the machine using a large number of real pattern samples. For feature extraction and dimension reduction two fundamental tools are usually applied, Principal component analysis (PCA) and discriminant analysis (DA). PCA and DA are used to apply eigenvector decomposition on the covariance matrices to disengage featured and therefore to find the uncorrelated features which are the most important in some senses [8].

2.2 Features Detection for Tracking

The Object tracking based on building paths by linking detections regions with related objects over the frames of the sequences as soon as the detector process gives the detections. In general, the process of defining an instance of real-world objects such as faces, trees, and furniture is called object detection. Typically, these algorithms depend on extracted features and learning algorithms to detect items. Different approaches are proposed to detect objects. Most of them depend on detecting a distinctive part or specific behaviour of objects such as colour, skin, feature, motion, or even any part of the human body. The technique that aims to detect a unique feature of objects in an image is called feature detection. This process enables experts to extract the objects within many statuses like different images, rotation, and scales [8]. Authors in [9] compute the histogram of local gradients around the extracted interested patches and draws boxes in a vector of 128 value. In [10], principal components analysis used by scale-invariant feature transform (PCA-SIFT) uses a vector of 36 value which is fast but less distinctive than scale-invariant feature transform (SIFT) as in the comparative study presented by [11]. Speeded-Up Robust Features (SURF) algorithm proposed in [12] is considered one of the robust and fast detector approaches. SURF uses a square shape filter to approximate the calculation of convolution second order Gaussian derivatives. The rapidly implemented SURF is resulted from implementing the convolution operations on an integral image. The entry of an integral image at a location $x = (x, y)^t$ is calculated by taking the summation value for each pixel of the original input image I at a rectangular area created by the origin and x. However, Herbert Bay claims in [12] that SURF outweighs the SIFT in addition to all previously proposed schemes in robustness, repeatability, and the speed of implementation.

For detecting human beings, face recognition represents a distinctive part of the whole body. This approach depends on defining the human face location and size with neglecting any other item in the scene [13]. The usual approach for detection face is based on the color of the skin area [14][15]. However, changing the color of the skin under the different conditions of the environment is one of many obstacles. One of the popular approaches to detect the human faces depends on matching facial elements, where it relies on the determination of certain parts of the human face such as eyes, mouth, and nose. In some cases, the detection of facial features is failed. This could happen for different reasons such as diversity of light, the image in tricky angular, and failing during the translation operation [15]. Thus, the processes of performance become slow or have errors in detection. Viola-Jones' algorithm is considered one of the robust and functional algorithms for fast analysis of images as well as faces detection [16]. This is because it depends on surrounded pixels that contain faces area within several rectangles. There have been three stages for the Viola-Jones algorithm. In the first stage, the Viola-Jones algorithm uses the value of the Haar feature selection algorithm to ensure that the extraction features belong to a face area rather than depending on the image's pixels directly. Haar feature algorithm provides a new method in image recognition that depends on turn on the original image to an intermediate representation named as "integral image". This converting contributes to speeding the calculation time to obtain the most feature evaluation. The Values of features are calculated by subtracting the sum number of pixels in white rectangles from the sum number of pixels in black rectangles. The integral image at any location composes of the summation of all pixels above and to the left of that location as in (1) [17].

$$ii(x, y) = \sum_{x' \leq x, y' \leq} i(x', y') \qquad (1)$$

Where $ii(x, y)$ refers to the integral image and $i(x', y')$ assigns to the original image. The second stage is Ada boost training, which works by detecting the strong and weak features. After that, it determines a specific weight to all features. Ada boost then uses the linear combination of weak classifiers to construct a strong classifier. This will contribute to minimize the learning difficulty and increase the effectiveness of the procedure [18]. Though, there are still thousands of features that need further filtering. Viola-Jones algorithm in the last stage implements a cascade of classifiers for more filtering of features. The basic idea is the cascade classifier consists of several strong classifiers and each classifier makes its judgment about whether the given region is a face area or not. Hence, the classifier which determined the given features as a non-face area will be neglected [19]. Another algorithm proposed in [20] works by extracting the facial pattern by matching the edge orientation map of the face model against the edge orientation map which is taken from the original image. In [21], however, a new fast algorithm is suggested depending on the information which is provided by the colour of the skin that orients the matching map.

The usual process for tracking uses state-space format which is based on a linear dynamic system [22]. This format characterizes the target object by the state of $\{x_k\}_{k=0,1,...}$, whose different states are over a dynamic time that is defined by $x_k = f_k(x_{k-1}, w_k)$, where the vector-valued f_k is a nonlinear function, and $\{w_k\}_{k=1,...}$ is a noise vector (mean Gaussian). The process model integrates with the measurement equation, that defines the relevance between each state, in addition to the relevance measurement at the specific time k as :$\{zk\}_{k=0,1...} = h_k(x_k + v_k)$, where z_k is the available measurements over k time, h is the nonlinear function and v is the noise vector. Each of $\{w_k\}$ and $\{v_k\}$ are assumed to be distributed in the same way and therefore they equal to zero. Tracking of an object over time is used to estimate the state x_k over all z_k measurements until the moment of the corresponding state, which can be found by the probability function $p(x_k \mid z_{1:k})$. In theory, the optimal solution is provided by recursive Bayesian filter which bases on two steps: prediction and update function. The first step uses the dynamic calculation paired with a probability at time $t = k - 1$ $p(x_{k-1} \mid z_{1:k-1})$. To predict the previous probability state at a prior time, the measurement of $p(x_k \mid z_{1:k-1})$ is used. Then, the next step uses the probability function of the current measurement $p(x_k \mid z_k)$ to find the backward ones $p(x_k \mid z_{1:k})$. Kalman filter is considered the optimal chosen in the case of the Gaussian noise and liners functions h_k , f_k. While the Extended Kalman filter (EKF) and Unscented Kalman filter are the optimal solutions in the Gaussian random noise and nonlinear functions h_k, f_k [23].

In addition, when the state of space is divided and has a limited number of states, the Hidden Markov Models filter is the best selection to overcome this issue [24]. This filter assumes that the tracking of the next state mainly depends on the current state. In fact, tracking multiple objects is a challenging process. The authors of [25] used several cameras to track many people. They were able to overcome the miss-detection problems and to detect the correct track by depending on homography transformation. It is possible to use Vector Kalman to track targets [26]. Moreover, the graph representation was also proposed for tracking multiple objects [27]. Usually, tracking operation shows error or misdetection problems, which are mostly due to the presence of noise in images, lighting difference, and/or in the case of presence a complexity in the motion. Such other trouble that rises a similar fail is due to the projecting a 3D real world on a 2D image, which could lead to losing some of the information as a result of the error in the tracking process. Different handling methods are used to reduce these misdetection problems like restricting object motion or simplifying it, and/or through using multiple tracking filters for each object [28].

2.3 Spatial relationships Applications

Several techniques have been proposed for detecting the spatial relationship. In this section, some of these approaches that are most relevant to the current study are considered. One of the earliest studies that discussed the encoding protocols of the special relations among objects in a picture was done by Freeman. He presented two types of relationships between objects; the first relation described the difference between objects' properties (shape, size, surface). While, the second suggestion was focused on identifying the spatial relationship, which was done by comparing the positions of objects like above, far, to the right, etc [29]. Another different approach was proposed by calculating the direction and weight of azimuth to detect the type of spatial relationships [30]. Centroid Method was presented in [31], which was assumed for comparison between angles and horizontal axis to obtain the final decision. Another method was based on co-occurrence triplets of objects; where, Triangle Spatial Relationships (TSR) was used to describe the image database as well [32]. These triplets should have geometric relationships encoded by using the angle of tringles that is shaped by objects. Besides that, minimum bounding rectangles in an approach called DISIMA were used to model the type of spatial relationship, which was carried out to distinguish image content in each notable object [33]. Identifying spatial relationships was also presented and described depending on Fuzzy Mathematical Morphologies [34]. Guadarrama et al [35] studied the ability to use visual and spatial information to find spatial relations. From another perspective review, hypergraph-based segmentation was implemented to define some of the spatial relationships in 2D images [36].

3. HYPERGRAPH APPROACH ADOPTATION FOR SPATIAL RELATIONSHIPS REVEALING

3.1 Objects Bordering

As aforementioned, the objects tracking process requests in the first place a detection process. The detectors' approach was implemented either on every frame or as soon as the target objects were displayed [37]. The process of information extraction from a single frame is a common approach for object detection. To prevent or reduce the error that might appear during the object detection process, temporary information extracted from a sequence of frames was utilized for identifying object area. The object detection task was implemented as soon as the human body had appeared in the video. The suggested method by Guo et al. [38] was adopted here, which was carried out for detecting the target object by using background subtraction based on Gaussian mixture models (GMM). The Gaussian algorithm represents the value of a specific pixel at a timeline known as the "pixel process", which represents the timing chain of the pixel value. By depending on a mixture of K Gaussian distributions, GMM can model the history of each pixel $\{X_1, \ldots X_t\}$, where X_t identifies the value of a pixel at time t. The probability of an observing given pixel was founded by equation (2):

$$p(X_t) = \sum_{i=1}^{k} \left(w_{i,t} * G(X_t | v_{i,t_{i,t}}, \Sigma_{i,t}) \right) \tag{2}$$

Where $w_{i,t}$: is the weight estimation, $v_{i,t}$ is the mean value, and $\Sigma_{i,t}$ is the covariance matrix of the *ith* Gaussian distribution for the mixture model at a given time, t and G are the functions of the Gaussian probability density represented by equation (3):

$$G(X_t | v_{i,t_{i,t},\Sigma_{i,t}}) = \frac{1}{|\Sigma_k|^{1/2} 2\pi^{n/2}} e^{\frac{1}{2}(x_t - v_k)^T} \Sigma_k^{-1} (x_t - v_k) \tag{3}$$

Where n is the dimension of X_t vector. Moreover, for a computational reason, the covariance matrix was motivated to be independent and it takes the following form equation (4):

$$\sum_{kt=} \sigma_k^2 I \tag{4}$$

Here, the motion of objects was solely used to detect the predicted tracking of the related object. For this task, Kalman filter (KF) was applied to estimate the body motion by using the obtain information from objects. Kalman filter is an estimator that can provide wide information about the object's location over each step of the time. In this case, the KF equation was calculated by equation (5) and the measurement equation was found by (6).

$$x_k = f x_{k-1} + w_k \tag{5}$$

$$z_k = H x_k + n_k \tag{6}$$

The F and H are called system matrix and measurement matrix respectively. The Kalman filter performs estimation over two steps: prediction and updating. It was applied therefore to predicate the object's location in each frame. The smaller the noise, the optimal the implementation of KF is reached, but some errors can happen when the detection and tracking processes record failure in the task, or in the case of the presence of noise in the current frame. Maintenance this error is essential for the system of the current study. Extra filter, which is the Viola-Jones algorithm, was applied to guarantee the reality of human body detection, especially the upper part of the body (face and shoulder), which had filtrated by KF. This step, which identifies the human body and neglects other objects, is necessary to prevent the conflict in the detection of spatial relationships between the human body and the silent object. The tracking window of each moving object was narrowed to surround a specific part of the human body. This does not just help to an accurate definition of spatial relationships, but it decreases the processing time and enhances the speed of operations [39]. In addition, it gives a clear view of users. The box of the human body is narrowed when it is far away from the camera. This leads to knowing that human bodies move far away from the camera and vice versa.

3.2 Detection of silent objects

To detect the non-moved object, a chair was used here, the Computer Vision System Toolbox, particularly the *detectSURFFeatures* detector function, was applied. This algorithm can detect the interesting point(s) in any invariant rotation. The order of implementation was feature detection, extraction, and then matching. The SURF algorithm detects objects by defining feature points first and then matching between similar points.

SURF can be partitioned into three steps [40]. The initial step determines the interest points of an image I at ant any point x and any angle σ by using Hessian matrix $H\ (I, \sigma)$ in equation (7).

$$H(I,\sigma) = \begin{vmatrix} I_{xx}(x,\sigma) & \cdots & I_{xy}(x,\sigma) \\ \vdots & \ddots & \vdots \\ I_{xy}(x,\sigma) & \cdots & I_{yy}(x,\sigma) \end{vmatrix} \tag{7}$$

Where $I_{xx}(x, \sigma)$ represents the convolution of Gaussian second-order derivatives in the image I and point x. It can refer Dxx, Dxy, and Dxy for approximations of second-order Gaussian derivatives, the determinant for Hessian could be found by (8).

$$det(H_{approx} = D_{xy}D_{yy} - (0.9D_{xy})^2) \tag{8}$$

In the initial step, the detected rectangular regions of interest points are passed to the second step where many methods are applied to determine only the major points. The goal of the second step is to provide a unique description of these interesting points. In this step, the interesting regions are split into four sub-regions. Then, the Haar wavelet response is calculated over each of the sub-regions with each of x and y direction at the interesting regions. After that, weighting the wavelet response with the Gaussian filter to increase the robustness toward the detected error process is conducted. Finally, the Haar wavelet responses are summed in both directions d_x and d_y over each of the sub-regions as shown in (9).

$$V = (\Sigma d_x + \Sigma d_y + \Sigma|d_x| + \Sigma|d_y|) \tag{9}$$

The last step is implemented by executing a comparing operation between features to detect the identical pair of features between different images [40]. Here, once the system is run, both images that include silent object and cluttered scene (frame from the video) are read in a grayscale image type. The $detectSURFFeatures$ function will be implemented on the images. Object points information will be returned by this function. The next step was carried out by applying the $extractFeatures$ function on interest points of each object's image and frame form to obtain descriptive information of each, which was used later for each image to implement the matching function. This function returns a matrix of corresponding features between each of the scene's features. The process of "Estimate Geometric Transform" was applied after that to implement the transformation process related to the identified important points, any other non-described points will be neglected. The bounding box of the fixed object is determined and transformed to the matched coordinates in the main scene (frame), which contains the fixed target object, to contour it. Establishing a rule that the chair is a stable non-moved object was set up; thus, the determination process of its position will be implied only once. The object information regarding the position was saved in the system and used later in the final step to define spatial relations, which has a great advantage by increasing the speed of the implementation process. If the position of the chair is changed the system will need a very trivial adjustment in order to repeat the determination process of the new object location.

3.3 Spatial Relationships Identification

Depending on geometric inferences, special relationships are detected. To achieve the aim of the program, each object should have the following parameters:

1) Ch and Co, the gravity center of human (h) and object (o), equations (10) and (11);

$$human\ box\ Ch(x,y) \;=\; (h(x) + human\ width/2, \; h(y) + human\ height/2) \tag{10}$$

$$object\ box\ Co(x,y) \;=\; (o(x_1) + (x_2 - x_1/2), \; o(y_2) + (y_4 - y_1/2)) \tag{11}$$

The bounding box of human is a matrix of 1*4, where the first two values represent the top_left (x, y) coordinate, while the third and the fourth values define the width and height of the box, that is used to find the gravity centre of human. The bounding box of the object is a polygon of five points, the gravity centre of the object can be found by making a subtracted operation.

2) The next parameters are the width and the height (W and H), of the bounding boxes of the human and the object. As aforementioned the width and the height of the human box can be found from its matrix. The width and the height of the object can be calculated by the following equations, (12) and (13) respectively:

$$object\ box\ W \;=\; x2 - x1 \tag{12}$$

$$object\ box\ H \;=\; y4 - y1 \tag{13}$$

3) $Uedge$ assigns to the deep of boxes according to the field of view. For the object, it is assigned to the value of $o(y3)$ as $Uedge\_o$. For the human, it is obtained by finding the result of the summation between human box's Y-coordinate and box's height as in (14):

$$Uedge_h \;=\; \text{h(y)+h(H)} \tag{14}$$

4) Wd is the subtracted result in width between the human box and the object box, equation (15);

$$Wd = h(W) - o(W) \tag{15}$$

5) $Wmax$, the maximum width which could be either human or object; and finally, **6)** $Hmax$, the maximum height between the human and the object boxes which also might be human or object. Figure 1 defines these parameters.

The spatial relationships that can be detected by the program are "In front of", "Behind", "Besides", "On" and "Under". Some of them are sub-classified to different partial relationships such as "In front to the right", "In front to the left", and "Direct in front of the object". Here, a human and chair (object) comparison was used. The human located "In front to the left" side of the chair was determined by the following comparison (16):

$$\begin{aligned}&(Ch(X) > Co(X))\&(Ch(X) - Co(X) > Wo/2)\&\\&(wh/1.5 > Wo)\end{aligned} \tag{16}$$

The human standing in front and to the right side of the chair was detected by (17):

$$\begin{aligned}&(Ch(X) > Co(X))\&(Ch(X) - Co(X) > Wo/2)\&\\&(wh/1.5 > Wo)\end{aligned} \tag{17}$$

Both of the above formulas to some extent are similar, except the value of X-coordinate is belonged to the center of the human box, which is either bigger or smaller than the X-coordinate of the object box. As a result, it is possible to distinguish the human position whether it is "To the left" or "To the right" side of the silent object (chair). In contrast, the "On" and "Under" relationships can be estimated depending on the Y-coordinate value. The spatial relationship of "On" was detected using the following comparison (18):

$$\begin{aligned}&(Ch(Y) > Co(Y))\&(Ch(X) - Co(X) > Wo/2)\&\\&(Uedge\_h(y) < Uedge\_o(y))\&(Wd < Wmax/4)\end{aligned} \tag{18}$$

Similar to the previous comparison, the "Under" relationship can be written using the following part $(Ch(Y) > Co(Y))$ instead of $(Ch(Y) < Co(Y))$ of the "On" formula. The first part of the comparison was used to determine the human box location whether it is located above or beneath the chair; the second and third parts however are required to distinguish the human box location whether it is located at the right or left side of the chair. The last part of the comparison, which is $(Wd < Wmax/4)$, is required to validate the distance between the human box and the chair, which should be very close. The human box located "To the left" side of the object can be detected using the following comparison (19).

$$\begin{aligned}&(Ch(Y) > Co(X))\&(Ch(X) - Co(X) > Wo/2)\&\\&(Wd < Wmax/4)\end{aligned} \tag{19}$$

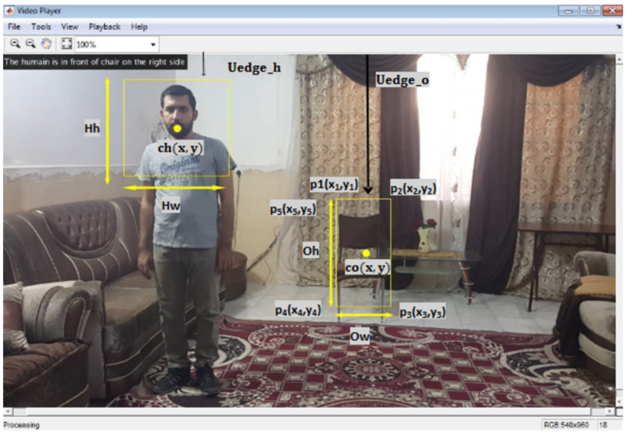

The humain is in front of chair on the right side

Fig.1: *The parameters used to identify the spatial relationships.*

The "To the right" side spatial relationship can be determined by only inverting the greater than symbol (>) of the previous formula to the less than symbol (<) as following: $(Ch(X) < Co(X))$. By using the second part of the aforementioned formula (19), $(Ch(X)–Co(X) > Wo/2)$, no overlapping between the boxes of the human and the object can occur. The human box in this case will never detect the object and can be located either on or under the object. Another restriction was used to determine the relation type. Writing the following comparison $(Wd < Wmax/4)$ is required to verifying that both boxes of the human body and the object should be almost similar in size, which is occurred only when both of them are located beside each other. The arrangement of the aforementioned comparisons was not randomly typed. By reading the first part, a general instruction will suggest that the human location is either on or beside the object, the second part however is restricted in turn to the first part; where if an overlapping of both boxes occurs, it means that the human location is on the object, the program in this case will neglect the other choice. By implementing the last comparison, both the human body and the object will be surrounded by approximately similar boxes in size in case they were beside each other. The "Behind" relationship can also be implemented by using different comparisons. For detecting the human behind and to the left side of the chair the formula (20) was carried out, while the formula (21) was applied to detect the human "Behind and to the right side" of the chair.

$$\mathbf{(Ch(X)>Co(X))\&(Ch(X)\text{-}Co(X)>Wh/5)\&(W0/1.3>Wh)} \tag{20}$$

$$\mathbf{(Ch(X)>Co(X))\&(Ch(X)\text{-}Co(X)>Wh/5)\&(W0/1.3>Wh)} \tag{21}$$

Restriction of the human direction, “To the left” or “To the right” side of the object, was achieved by implementing the first comparison $(Ch(X) > Co(X))$; the rest of the comparisons are in turn responsible for restricting the human location which should be always behind the object, the size of the object box, in this case, is bigger than the human one. In the case of the “Direct behind relationship”, the comparison (22) was executed.

$$(ChY<CoY)\&(Wo/1.3>Wh)\&(ChX\text{-}CoX<Wh/5) \tag{22}$$

This part (ChY<CoY) of the comparison is implemented when the level of the human box is higher than the object box. The second part (Wo/1.3>Wh) is only implemented based on the box size, when the human location is behind the object the human box will always be smaller in the size, regardless of the direction whether it is on the right behind, left behind, or central behind. In the contrast, the last part of the comparison (ChX – CoX<Wh/5) is only implemented when the human is directly behind the object, the boxes of the human body and the object, in this case, are vertically identical. To establish a relationship of the human body between two entities (such as a chair and a table), the spatial relationship “Between” in this case can be applied. It can be estimated by testing the human position on the left side of the chair with the right side of the table (left chair_ right table) or vice versa (left table_ right chair) by applying both the equation No. (16) & (17). For all the previous relationships, the used values (2, 1.5, 1.3) are smoothing values that were chosen after several testing trails of other values, the above values represent the threshold of the truth value. The appendix (1) and Figure (2) explains the procedure of the system briefly.

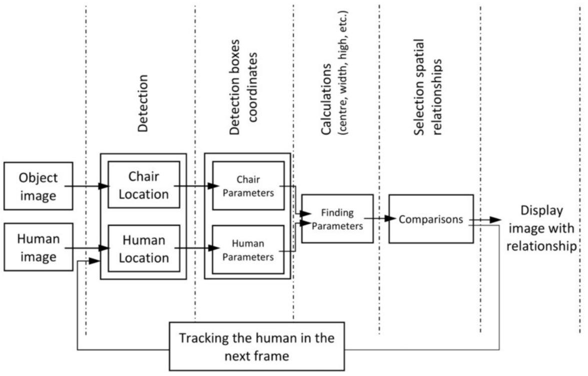

Detection
Detection boxes coordinates
Calculations (centre, width, high, etc.)
Selection spatial relationships
Object image
Human image
Chair Location
Human Location
Chair Parameters
Human Parameters
Finding Parameters
Comparisons
Display image with relationship
Tracking the human in the next frame

Fig.2: *The system action steps for image analysis and defining the spatial relationship.*

4. EXPERMINTAL RESULTS

The results show the system’s validity of prescribing spatial relationships among different related objects. Here a motion-based object tracking was modulated in order to determine and extract the type of spatial relationships of objects in a complex streaming video. A subset of frames as shown in (Figure 3) shows that the detection of the relationship type is mainly dependent on the accuracy of surrounding objects within the right size of the bounding box. When one of the boxes is bigger than the other one, for example, it means that the object enclosed with the bigger boxes is in front of the other object. The results of the system performance are presented in Table 1.

A group of frames to discern the effectiveness of the algorithm was demonstrated. The used algorithm is dependent mainly on the location of the centroid points and the size of frames. In Figure 3A for example, the relationship of “On” is represented. Both boxes of the human body and the chair are surrounded by individual frames, both frames approximate by the length of the frame width, the system as a result was able to distinguish that their centroid points were approximate regarding the X axis. There is no big distance between the frames, thus, the relationship was recognized as “On” spatial relationship. The system was learned that the horizontal spacing rate between the two frames should be very small. In the case of large distances, there is a possibility of appearing a system error. The ratio of distances between both frames was determined, where its value should not exceed the quarter value of the largest frame. The error may appear in case the frame of the human body was recognized as a moving object, which has not set yet on the chair, as shown in (Figure 3B). Where the system was able to recognize the human face and the chair object, but the relationship was not determined. This is because the Viola-Jones algorithm was not able to recognize the presence of the human body due to the blurred facial features.

In (Figure 3C), because the human was very close to the camera, he was determined with a frame that was bigger than the chair frame. Moreover, because the X axis of the human is less than the chair X axis the system was able to determine the relation as “In front of on the right”. In Figure 3D, the X axis of the centroid points of both frames, which were very approximate to each other, is shown. The frame width of the human was much less than the chair frame, in addition to the horizontal distances between the frames, which were too big. Thus, the system was able to recognize this relationship as “Direct behind of”. However, the system may not able to distinguish this relation if the human directly moves to the chair, where both frames are being in the same size. The system can recognize the “Behind of on the left” relationship, if the human frame size is less than the object frame size, and if the X axis of the human centroid point is greater than the centroid point of the chair (Figure 3E). The different sizes of both frames and the larger value of the centroid’s X-coordinate of the chair allow the system to distinguish the relation as “Behind chair on the left” (Figure 3F). The relation of “Between” can be detected as long as the human location is between the two objects (the chair and the table). After detecting the location of the table, it is possible to compare its location with all the previous comparisons related to the chair. Thus, each of the “To the right” and “To the left” was applied to detect the “Between” relation. Figure 4 shows it as well as more displaying for the previous relationships.

5. IMPLEMENTATION ISSUES

The system was implemented on the Microsoft Windows 7 Professional 64-bit operating system using Matlab programming language in addition to the extensive use of the Mathworks library [43], [44]. Processing time varies, which took about 3 seconds to process 1 frame of a 3-minute video. Some aspects must be taken into account when recording videos. A good source of light will enhance the detection process. These issues of light can be addressed with an efficient camera combined with an experienced photographer. Here, the location of the object was determined only once, and its location was saved for the rest of the other spatial comparisons. It is imperative that the camera is well stabilized when filming. Any vibration of the camera leads to a possible appearing of problems in the synchronization process. The main reason belongs to the longer time spent by the system on processing the frame and identifying the silent object. This may cause a frameshifting away from the object. However, the treatment of this issue was achieved by repeating the detection of the silent object with each frame, this might slow down the processing. Figure 5 shows the variety of times to process the same video. It took 163.407469 seconds to process 467 frames by repeating the chair detection process with each frame, whereas 141.544805 seconds were required when adjusting the detection only for once.

Table 1: *System validity test outcome regarding each type of spatial relationships.*

Spatial relations	Test No.	Fail No.	% of correction	The reason for the error
On/Above	5	1	80	Human intended to be above the object
Beside on to the left	10	0	100	-
Beside on to the right	10	1	90	Failing in the detection process of the human face
Behind	7	2	71.4	No distance was between the chair and the human body, or the human was in front of the object
Behind on to the left	7	2	71.4	Having the wrong human box (the detection was for the whole body), or the human was on the left side
Behind on to the right	7	2	71.4	The human was close to the chair, or the diversity of the light caused a failing in the process of the detection
Direct In front of	10	1	80	No distance was between the object and the human body
In front of to the left	10	1	90	The human was very close to the chair.
In front of to the right	10	2	80	Diversity of the light caused a failing in the process of the detection, or the camera used to record the video was not stable
Between	5	0	100	
Overall	81	12	83.4	

6. DATA SET

I tested the system on videos that were recorded using the camera of the Samsung Galaxy Note 5 mobile, these videos differ in size but most of them are not too long, such as, a video of 45.1 MB. Most videos were recorded with minimal vibration, which was happened due to the shaking of the hand during the recording. The majority of the videos have more than two relationships, thus the number of the tested videos is 24.

7. CONCLUSION

The image segmentation process used for retrieval an essential information required for different purposes has been involved in many applications. Several modern programming approaches and machine learning algorithms have brought great opportunities for building such efficient applications. The work presented here provides a reliable, feasible method to detect spatial relationships between a human being and a silent object near to him. Successful usage of motion-based object tracking that combines three already known algorithms was the key to achieving the study’s aim. Which was achieved after implementing the following: the GMM algorithm was used in the first place to detect the human body; Whereas, the

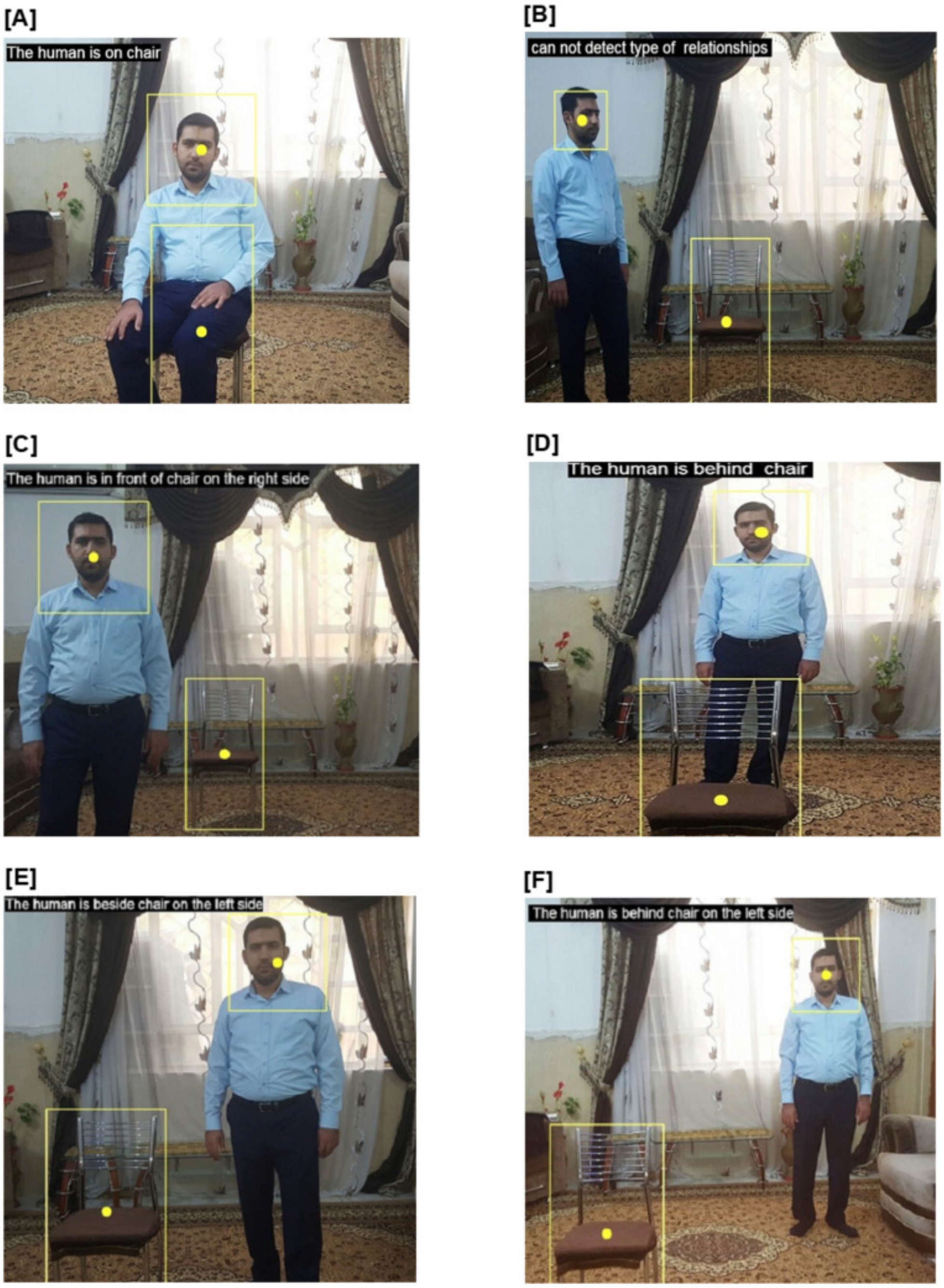
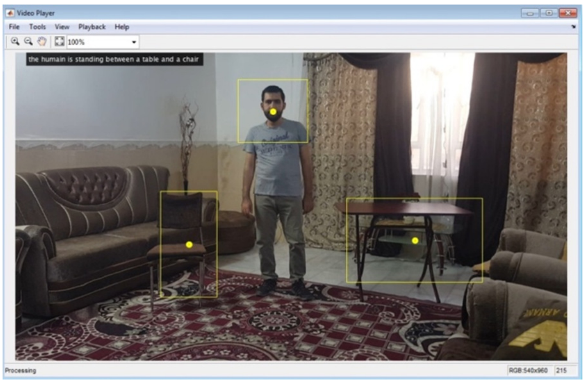

[A]
The human is on chair
[B]
can not detect type of relationships
[C]
The human is in front of chair on the right side
[D]
The human is behind chair
[E]
The human is beside chair on the left side
[F]
The human is behind chair on the left side

Fig.3: *This figure is defining the spatial relationships between a moved human and a stable object (chair) detected in a 3D scene. The figure shows the following detected relationships; On (Panel A), In front of on the right (Panel C), Direct behind (Panel D), Beside on the left (Panel E), and Behind on the left (Panel F). Panel B shows undetectable type of relationship due to the inability of the system to detect the man properly..*

algorithms of the KF and the Viola-Jones, which minimize the KF error, were utilized for tracking the detected object. Detection of the fixed object, in addition, was achieved by using SURf approach. Finally, for defining the spatial relationship between the detected areas, Hypergraph-Object Oriented structure was applied. The geometric inference was used, here, for determining several spatial relationships. This was done by defining the human face with a box as same as for defining the other silent object(s). The system was established to compare the coordinates of these boxes in order to define the spatial relationships. Although the simulation results, here, showed that the system was able to define several spatial relationships between a moving object and another stable one at an efficient successful rate, system performance can be improved to a higher level. Incorporating other features may improve object detection and reduce the system's errors. More information about the environment of the related image/frame could be gathered and planted, which adds more geometric inference for efficient and accurate detection of further spatial relationships. To sum up, the idea of this study is indeed establishing the basis for more sophisticated studies in the future. More complex spatial relationships between several moving entities could be developed, which are very required for different commercial and scientific purposes.



the humain is standing between a table and a chair

Fig.4: The "Between" spatial relationships.

ACKNOWLEDGMENT

This work was funded by the Department of Computer Science at the Faculty of education and the Department of Urban Planning (University of Kufa-Iraq), project grant NNA00088. I would like to thank Dr. Bassam K. Kudhair at the Faculty of Science (University of Kufa) for carefully revising and proofreading of this paper.

CONFLICT OF INTEREST

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

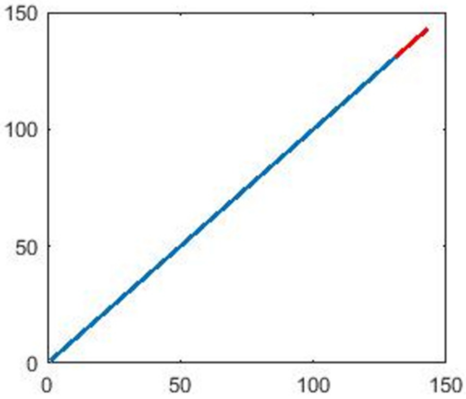


Fig.5: Reduction results of the processed time.

References

[1] T. Jaworski, J. Kucharski, "The use of fuzzy logic for description of spatial relations between objects," *Autom. / Akad. Górniczo-Hutnicza im. Stanisława Staszica w Krakowie. T. 14*, pp. 563–580, 2010.

[2] B.C. Condé,S. Fuentes, M. Caron, D. Xiao, R. Collmann and K.S. Howell, "Development of a robotic and computer vision method to assess foam quality in sparkling wines," *Food Control. 71*, pp. 383–392, 2017.

[3] E. Ganea and M. Brezovan, "An hypergraph object oriented model for image segmentation and annotation," in *Proceedings of the International Multiconference on Computer Science and Information Technology*, pp. 695–701, Wisła, Poland, 2010.

[4] E. Ganea and M. Brezovan, "Image indexing by spatial relationships between salient objects," in *2011 Federated Conference on Computer Science and Information Systems, FedCSIS 2011*, pp. 699–704., Szczecin, Poland, 2011.

[5] J. Qiu, Q. Wu, G. Ding, Y. Xu and S.Feng, "A survey of machine learning for big data processing," *EURASIP Journal on Advances in Signal Processing 2016* vol.1, pp.1-16, 2016.

[6] C. Rudin and K.L. Wagstaff, "Machine learning for science and society," *Mach Learn.*, vol.95, pp.1–9, 2013.

[7] F. Tango and M. Botta, "Real-Time Detection System of Driver Distraction Using Machine Learning," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 894-905, June 2013.

[8] J. Pedersen, SURF: Feature detection & description, 2011.

[9] E. Mönestam and A. Behndig, "Impact on visual function from light scattering and glistenings in

intraocular lenses, a long-term study," *Acta Ophthalmol*, vol.89, pp.724–728, 2011.

[10] Yan Ke and R. Sukthankar, "PCA-SIFT: a more distinctive representation for local image descriptors," *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2004. CVPR 2004., pp. II-II, 2004.

[11] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615-1630, October 2005.

[12] H. Bay, T. Tuytelaars and L.Van Gool, "SURF: Speeded up robust features," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, pp. 404–417, 2006.

[13] L.Shen and L. Bai, "Face Detection in Grey Images Using Orientation Matching," in *Proceedings 17th European Simulation Multiconference*, pp. 2–7, 2003.

[14] D. Karmakar and C.A. Murthy, "Face recognition using face-autocropping and facial feature points extraction," *PerMIn '15: Proceedings of the 2nd International Conference on Perception and Machine Intelligence*, pp.116–122, February 2015.

[15] B. Dhivakar,C. Sridevi,S. Selvakumar and P. Guhan, "Face detection and recognition using skin color," *2015 3rd Int. Conf. Signal Process. Commun. Networking, ICSCN 2015*, pp.3–9, 2015.

[16] P. Viola and M.J. Jones, "Robust Real-Time Face Detection," *International Journal of Computer Vision*, vol.57, pp.137–154, 2004.

[17] H. Jia, Y. Zhang, W. Wang and J. Xu, "Accelerating Viola-Jones Facce Detection Algorithm on GPUs," *2012 IEEE 14th International Conference on High Performance Computing and Communication & 2012 IEEE 9th International Conference on Embedded Software and Systems*, pp. 396-403, 2012.

[18] B. Sun, S.Chen, J. Wang and H. Chen, "A robust multi-class AdaBoost algorithm for mislabeled noisy data," *Knowledge-Based System*, vol.102, pp. 87–102, 2016.

[19] L. Cuimei, Q. Zhiliang, J. Nan and W. Jianhua, "Human face detection algorithm via Haar cascade classifier combined with three additional classifiers," *2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI)*, pp. 483-487, 2017.

[20] B. Fröba and C. Külbeck, "Real-Time Face Detection Using Edge-Orientation Matching," *3rd International Conference on Audio and Video Based Biometric Person Authentication*, pp.78-83, 2001.

[21] L. Bai and L. Shen, "Face Detection by Orientation Map Matching," in *International Conference on Computational Intelligence for Modelling Control and Automation*, pp. 1–7, 2003.

[22] Q. Ge, T. Shao, C. Wen and R.Sun, "Analysis on strong tracking filtering for linear dynamic systems," *Mathematical Problems in Engineering*, vol.2015, 2015.

[23] A. Nakabayashi and G. Ueno, "An extension of the ensemble kalman filter for estimating the observation error covariance matrix based on the variational Bayes's method," *Monthly Weather Review*, vol. 145, pp.199–213, 2017.

[24] O.O. Ogundile, A.M. Usman and D.J.J. Versfeld, "An empirical mode decomposition based hidden Markov model approach for detection of Bryde's whale pulse calls," *The Journal of the Acoustical Society of America*, vol.147, pp.EL125-EL131, 2020.

[25] J. Dias and P.M. Jorge, "People tracking with multi-camera system," *Proceedings of the 9th International Conference on Distributed Smart Cameras*, ICDSC '15, pp.181–186, September 2015.

[26] S.A. Vigus, D.R. Bull, C.N. Canagarajah, "Video object tracking using region split and merge and a Kalman filter tracking algorithm," *Proceedings 2001 International Conference on Image Processing*, vol.1, pp.650–653, 2001.

[27] G. Medioni, I. Cohen, F. Bremond, S. Hongeng and R. Nevatia, "Event detection and analysis from video streams," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 8, pp. 873-889, August 2001.

[28] S.E. Li, G. Li, J. Yu, C. Liu, B. Cheng, J. Wang and K. Li, "Kalman filter-based tracking of moving objects using linear ultrasonic sensor array for road vehicles," *Mechanical Systems and Signal Processing*, vol.98, pp.173–189 2018.

[29] J. Freeman, "The modelling of spatial relations," *Computer Graphics and Image Processing*, vol. 4, pp.156–171, June 1975.

[30] H. Yan, Y. Chu, Z. Li and R. Guo, "A quantitative description model for direction relations based on direction groups," *GeoInformatica*, vol.10, pp.177–196, 2006.

[31] R. Krishnapuram, J.M. Keller and Y. Ma, "Quantitative Analysis of Properties and Spatial Relations of Fuzzy Image Regions," in *IEEE Transactions on Fuzzy Systems*, vol.1, no.3, pp.222–233, August 1993.

[32] N.V. Hoàng, V. Gouet-Brunet, M. Rukoz and M. Manouvrier, "Embedding spatial information into image content description for scene retrieval," *Pattern Recognit*, vol.43, pp.3013–3024, 2010.

[33] V. Oria, M.T. Ozsu, P.J. Iglinski, B. Xu and L.I. Cheng, "DISIMA: An object-oriented approach to developing an image database system," *Pro-*

ceedings of 16th International Conference on Data Engineering, pp.672–673, 2000.

[34] I. Bloch, "Fuzzy spatial relationships for image processing and interpretation," *Image and Vision Computing*, vol.23, pp.89–110, 2005.

[35] S. Guadarrama et al., "Grounding spatial relations for human-robot interaction," *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1640-1647, 2013.

[36] S. Jouili, S. Tabbone, "Hypergraph-based image retrieval for graph-based representation," *Pattern Recognit*, vol.45, pp.4054–4068, 2012.

[37] A. Yilmaz, O. Javed, M. Shah, "Object tracking: A survey," *ACM Computing Surveys*, vol.38, pp.1–45, 2006.

[38] T. S. F. Haines and T. Xiang, "Background Subtraction with DirichletProcess Mixture Models," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 4, pp. 670-683, April 2014.

[39] X. Li, K. Wang, W. Wang and Y. Li, "A multiple object tracking method using Kalman filter," *The 2010 IEEE International Conference on Information and Automation*, pp. 1862-1866, 2010.

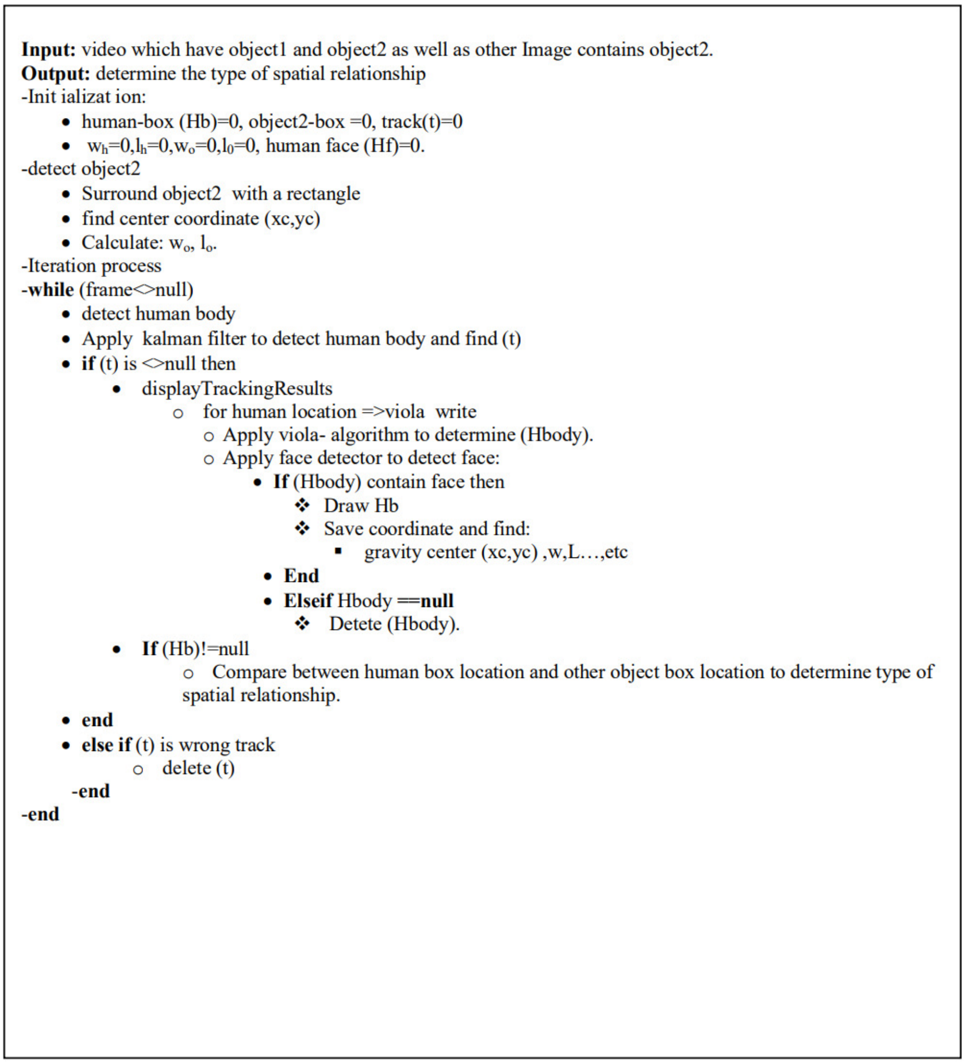
[40] M. Muthugnanambika and S. Padmavathi, "Feature detection for color images using SURF," *2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)*, pp. 1-4, 2017.

[41] The MathWorks Libraries, Available from: &lt; `https://www.mathworks.com/discovery/face-recognition.html\&gt;`

[42] The MathWorks Libraries, Available from: &lt; `https://www.mathworks.com/help/vision/ref/detectsurffeatures.html\&gt;`

APPENDIX

Appendix 1: *The main steps of the system*



```
Input: video which have object1 and object2 as well as other Image contains object2.
Output: determine the type of spatial relationship
-Init ializat ion:
    • human-box (Hb)=0, object2-box =0, track(t)=0
    • $w_h$=0,$l_h$=0,$w_o$=0,$l_0$=0, human face (Hf)=0.
-detect object2
    • Surround object2  with a rectangle
    • find center coordinate (xc,yc)
    • Calculate: $w_o$, $l_o$.
-Iteration process
-while (frame<>null)
    • detect human body
    • Apply  kalman filter to detect human body and find (t)
    • if (t) is <>null then
          • displayTrackingResults
                o  for human location =>viola  write
                   o Apply viola- algorithm to determine (Hbody).
                   o Apply face detector to detect face:
                         • If (Hbody) contain face then
                               ❖ Draw Hb
                               ❖ Save coordinate and find:
                                    ▪ gravity center (xc,yc) ,w,L…,etc
                          • End
                          • Elseif Hbody ==null
                               ❖  Detete (Hbody).
          • If (Hb)!=null
                o  Compare between human box location and other object box location to determine type of
                spatial relationship.
    • end
    • else if (t) is wrong track
            o   delete (t)
     -end
-end
```

Noralhuda N. Alabid is a lecturer and researcher at the department of Computer Science, Faculty of Education, University of Kufa, Najaf, Iraq; She has an MSc in the field of Advanced Computer Science. She was graduated from the Department of Computer Science, faculty of Science, University of Sheffield, UK. Her background is in the field of computer science with special interests in Image processing, Bioinformatics, and medical statistics.