Article

Dynamic Queries through Augmented Reality for Intelligent Video Systems

Josue-Rafael Montes-Martínez [1,*], Hugo Jiménez-Hernández [2,*], Ana-Marcela Herrera-Navarro [2], Luis-Antonio Díaz-Jiménez [2], Jorge-Luis Perez-Ramos [2] and Julio-César Solano-Vargas [1]

[1] Center for Engineering and Industrial Development, Av. Playa Pie de la Cuesta No. 702, Desarrollo San Pablo, Queretaro 76125, Mexico; jsolano@cidesi.edu.mx

[2] Informatic Faculty, Autonomous University of Queretaro, Av. de las Ciencias s/n, Juriquilla, Queretaro 76017, Mexico; mherrera@uaq.mx (A.-M.H.-N.); luis.diaz@uaq.edu.mx (L.-A.D.-J.); jorge.luis.perez@uaq.edu.mx (J.-L.P.-R.)

[*] Correspondence: jrmontes@posgrado.cidesi.edu.mx (J.-R.M.-M.); hugo.jimenez@uaq.mx (H.J.-H.)

Abstract: Artificial vision system applications have generated significant interest as they allow information to be obtained through one or several of the cameras that can be found in daily life in many places, such as parks, avenues, squares, houses, etc. When the aim is to obtain information from large areas, it can become complicated if it is necessary to track an object of interest, such as people or vehicles, due to the vision space that a single camera can cover; this opens the way to distributed zone monitoring systems made up of a set of cameras that aim to cover a larger area. Distributed zone monitoring systems add great versatility, becoming more complex in terms of the complexity of information analysis, communication, interoperability, and heterogeneity in the interpretation of information. In the literature, the development of distributed schemes has focused on representing data communication and sharing challenges. Currently, there are no specific criteria for information exchange and analysis in a distributed system; hence, different models and architectures have been proposed. In this work, the authors present a framework to provide homogeneity in a distributed monitoring system. The information is obtained from different cameras, where a global reference system is defined for generated trajectories, which are mapped independently of the model used to obtain the dynamics of the movement of people within the vision area of a distributed system, thus allowing for its use in works where there is a large amount of information from heterogeneous sources. Furthermore, we propose a novel similarity metric that allows for information queries from heterogeneous sources. Finally, to evaluate the proposed performance, the authors developed several distributed query applications in an augmented reality system based on realistic environments and historical data retrieval using a client–server model.

Keywords: distributed systems; query by reference; augmented reality; metric approach; homogeneity; vision distributed system

1. Introduction

A distributed video surveillance system consists of interconnected cameras that share information in specific environments for monitoring the human activities, motion detection, behavior, and interaction between individuals through the tracking of moving objects [1–4], thus providing great versatility in obtaining information for analyzing the acquired data. One of the main purposes of the distributed camera system is to provide an automatic interpretation of the scenes and predict the interactions of the objects; however, there are different factors that can provide an automatic understanding of the scenes and predict the interactions of the objects. Some of the different factors to consider when acquiring information about a specific scenario are the characteristics of the cameras available, the position, and the number of cameras [5,6]. One of the significant factors to consider when

installing a video surveillance system is that, when increasing the number of cameras, the coverage area increases, and so does the amount of information to manipulate. Every day, we are exposed to different types of surveillance carried out with devices that constantly generate video data [2,7]. According to the analyses carried out by [8,9], the increase in information obtained by video surveillance devices has grown exponentially to the point that up to 100 GB in data is acquired for each video surveillance camera in a monthly period. Particularly, IntelliVision have deployed more than four million surveillance cameras [10]. In video surveillance systems, various challenges must be met to achieve the interconnection and exchange in information from one point to another through different functional units such as cameras, means of sending information, and storage units. With an increase in distributed systems, it is necessary to propose or apply architectures that help to achieve interoperability between the different existing elements, such as sensors and actuators, to facilitate the sending and reception of data. Some works, such as [11–15], describe different middleware architectures to carry out the exchange in data and use of information in addition to describing the elements that influence successful interoperability between different sources of information, such as the communication medium, programming techniques, software architecture, sensors, semantics, and adaptive control, among others.

The most common use of the information obtained from a distributed camera system is motion detection and the tracking of objects of interest [3]. Distributed camera systems have proven useful as a passive means of detecting events or situations of interest; for example, the detection of an atypical behavior [16–18], the evolution of the dynamics of movement of an object in time [19–21], target tracking for surveillance applications [3,22–27], and tridimensional reconstruction [28–30]. Other important works are [16,31–35]. These works have already shown that they are useful for recognizing events of interest in a defined context. However, it has been observed that the complexity of these approaches increases considerably when the local dynamics are very complicated [36] or when there is a distributed system of cameras [37]. Many authors have proposed different methodologies and tools to automatically or semi-automatically detect atypical events in a monitored area. Currently, the high quality and high rate of image acquisitions per second are more frequently used in video surveillance systems, for which increasingly innovative methods are being presented to find motion primitives that help to interpret the activities raised in the stage [38].

Due to the diversity of approaches that exist, the fusion of information is a complex task due to the parameters each one requires and the results each delivers. Table 1 shows a table that represents the heterogeneity in the results when applying different object-tracking methods [39].

Table 1. The different object tracking methods.

Methodology	Occlusion Tolerant	Detection Every Frame	Training Rule Required	Number/Type Objects Tracking
Simple Template Matching	Partial	No	No	One object
SVM	Partial	No	Yes	One object
Contour Matching	Yes	No	Yes	Multiple objects
Shape Matching	Yes	No	No	One object
Kalman Filter	Yes	Yes	No	Multiple objects
Particle Filter	Yes	Yes	No	Multiple objects
Mean Shift	Partial	No	No	One object

Most of the works focused on automated video surveillance can be split into two main approaches: (1) Surveillance system model techniques, where different approaches are concerned with the integration of various vision algorithms to build a completed

surveillance system; (2) Distribution, communication, and system design, where it is analyzed how such methods need to be integrated into large systems to mirror the needs of practical closed-circuit television (CCTV) installations in the future.

When applying methods to obtain the characteristics of interest from the data, the displayed information must have visual support when interpreting the patterns obtained. Visualization is the best way to convert a large amount of information into small spaces that allow a quick and efficient interpretation of results [40]. According to works such as [41], data visualization must undergo seven steps: acquisition, grammatical analysis, filtering, data mining, representation, and representation improvement and interaction (within the representation part). Works such as [42–45] indicate that visualization depends on the amount of data, the information to be displayed, the homogeneity or heterogeneity of the data, the representation medium, and the dimensionality of the data. A good visualization technique and the right technologies can turn abstract information into a visual representation that can be decoded more easily, efficiently, and meaningfully [46]. Currently, with the technological advances in recent years, new information visualization alternatives, called interactive technologies, have emerged. These seek to form a three-dimensional representation that can be captured within the real world through the projection of data, the creation of virtual scenarios that resemble what exists in the real world, and the display of information through conventional devices—such as cell phones or tablets and cellphone gadgets—which turn these technologies into tools where users can interact with the information [43,47–49]; an example of these interactive technologies are augmented and virtual reality systems.

In particular, augmented reality is a technology of great interest due to the representation of information. Augmented reality is an interactive technology that allows computer-generated three-dimensional objects to be added to the real world through a viewing device such as a monitor, a cell phone, or more specialized devices such as augmented reality glasses or headsets. This technology emerged in the late 60s [50,51]; however, it has been widely used for about a decade due to the technological advances that facilitate its visualization. One of the characteristics that arouses the most significant interest is the improvement in user perception and interaction with the real world [52]. Among the most essential characteristics of its development are the following [50]:

1. It combines the real and virtual world.
2. It is interactive in real time.
3. It is represented in 3D.

Since it is an interactive technology, it manipulates information from visualization in the real world, therbey representing the information of interest for a different interpretation of the data. Many applications can find uses in different sectors such as medicine [53], industry [54], education [55], video games [56], tourism [57], and marketing [58].

This work aims to show the feedback of the heterogeneous historical information stored in a data repository generated from a distributed camera system by applying different motion detection methods. The proposal of homogeneity of heterogeneous information for its comparison and visualization is given by the generation of a global space where feedback is achieved. This generates a reference trajectory obtained from the global space with the help of an augmented reality system, which is compared with the data repository allowing the proposal of a new metric to form queries from heterogeneous information. The results are then finally shown in real environments through augmented reality using a client–server model.

2. Methodology

The states of this work can be summarized as follows:

*   Detection of movement dynamics.
*   Common global space.
*   Query trajectories.

- Similarity analysis.
- Augmented reality for querying and visualization data.

2.1. Detection of Movement Dynamics

In the literature, several works have proposed different methodologies and tools to automatically or semi-automatically detect information of interest such as contours, trajectories, and characteristic extraction to represent objects of interest or abnormal behavior. For example, Hugo Jimenez et al. [32] proposed an unsupervised approach to detect the atypical behaviors of vehicles at intersections; in [16], the movement of vehicles was encoded as a set of finite states generating hierarchical rules that are associated with normal behaviors in a scenario; in [39], the authors described the basic methods for motion detection and object tracking in video systems in addition to describing the steps required to carry out this process. There are three steps for this: the detection of the object of interest (people, vehicles, birds, etc.), classification, and the tracking of the object. For object detection, the methods that are most frequently used in the literature are mentioned: image difference, background subtraction, and optical flow.

In distributed monitoring systems, it is common for each camera to have a particular method applied; this is identified as a M_i model, which helps to interpret the dynamics in specific scenarios. Each M_i model generates historical information $H_i = \{h_1, h_2, \ldots\}$; however, for a couple of models M_i, M_j, for an instant in time t, it is known that $h_t^i \neq h_t^j$. This means that the information is not comparable; therefore, searches or data queries require that an M_i model contain a T_i transformation. This transformation is defined as $dom(H_i) \leftrightarrow E$.

This work aims to show the feedback of the heterogeneous historical information stored in a data repository that was generated from a distributed camera system by applying different motion detection methods. Motion dynamics detection aims to represent objects of interest with a set of pixels on a sequence of images [59]. Table 2 shows some of the models widely used in the literature for detecting the movement dynamics of objects of interest [39,60].

Table 2. Models of detection of movement.

Model of Detection	Detected Movement	Description
Temporal differences	$\|I_n(x) - I_{n-1}(x)\| > T$	The moving object is determined by calculating the difference between two consecutive images.
Image average	$\varepsilon = \sum_x [F(xA + h) - (\alpha G(x) + \beta)]^2$	The moving object is determined by calculating the addition of images in every frame and dividing by the total number of images added.
Optical flow	$I_{m \times n} = (I_1 + I_2 + \ldots + I_n)/n$	Optical flow is to calculate the optical flow field of the image and perform clustering processing according to the optical flow distribution characteristics of the image.
Gaussian of mixture	$I(x, t) = \sum_{i=1}^{k} G_i(\mu_i, \sigma_i)$	This method is used when there are multi-modal backgrounds. This method proposes modeling each pixel as a mixture of multiple Gaussians.
Background model	$\|I_t(x, y) - B_t(x, y)\| > T$	This method compares a sequence of images with a reference image containing static elements; the reference image is called a background image.

It is known that, by applying an M_i model to the information provided by a camera, results can be obtained that allow feedback on what has happened or is happening in a specific scenario. Based on this idea, by combining the results of different M_i models, the information can be complemented to better interpret the dynamics generated by the objects of interest. The latter can only be achieved if a way is defined to homogenize the information from different models.

This work proposes a way through which to homogenize the information of different models, defining a global space where all the trajectories can be mapped regardless of the model used to obtain the dynamics of movement of objects of interest (the objects of interest in the models are people who are moved from one point to another within the viewing area of a distributed two-camera system). A different model is applied to each camera to detect movement dynamics, temporal differences [32], and Gaussian mixtures [61]. In particular, temporal differences is a model that is tolerant to light changes [32], whereas Gaussian mixture is a widely used model with static cameras [62]. The movement dynamics of the objects were stored in an independent data repository for each camera. Figure 1 shows the movement dynamics generated by the objects of interest. Only one part of each video captured by each camera was analyzed, whereas for Camera 1, the temporal differences model was applied, and 38 trajectories were generated. In contrast, for Camera 2, the Gaussian mixture model was applied, and 22 trajectories were generated. These trajectories that were used for the acquisition of information are expressed in pixels.

Figure 1. Trajectories obtained by each camera. Red dots represent movement dynamics, and colorful blocks represent matching areas.

2.2. Common Global Space

A common global space can be defined as where information becomes homogeneous, thus allowing for comparison and manipulation. Different ways through which to obtain equivalences exist in artificial vision systems with different workspaces. To obtain equivalences of the information provided by different cameras, a process that can be divided into two aspects must be generated. The first is through a calibration process where the intrinsic and extrinsic parameters of the cameras are considered; this also requires a pattern, or elements within the scenario, that allows for characteristics to be extracted, the same ones from which, by applying a transformation, equivalences are obtained between the information obtained by the cameras expressed in pixels and the real-world information commonly expressed in millimeters or centimeters. The second is to obtain equivalences without a calibration process. A linear transformation is applied through homography, which is based on the information extracted from the scenario that can be taken from active or passive marks.

In this work, it is necessary to define the global space and its units of measurement to create a unified space. The global space becomes defined in a global position and local orientation. The global position is given by the UTM coordinates of the building. The center of the building represents the origin. The UTM ID is then used to store the information in DB. Each camera becomes linked to this origin by the projection of a local homograph. As a linear transformation, the homograph becomes constructed by active color markers in the scenario. These markers are associated with all cameras used, and the colors are used to orientate the camera direction in the scenario. Then, all camera positions are indexed and stored relative to the origin.

The global space is expressed with two-dimensional coordinates. The transformation is carried out using the information obtained by each camera, which contains all the

information expressed in centimeters to make queries that can provide feedback through visualizations in augmented reality in the real world.

To obtain the equivalences between the 2D or 3D coordinates of two different spaces [63], a linear projective transformation is estimated using homography on a homogeneous vector of 3×1, which is represented by a non-singular matrix of 3×3.

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \times \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}. \tag{1}$$

In other words $x' = Hx$.

To map the spaces generated by each camera to the common global space, several common points, which were defined as active marks, were installed (rectangles: red, yellow, green, and blue), as shown in Figure 2.

Figure 2. The perspective of each camera. Colorful blocks are reference marks necessary for the construction of the global space.

A representation of the distributed system and the common points is shown in Figure 3, where the colorful blocks represent the active marks called points of interest.
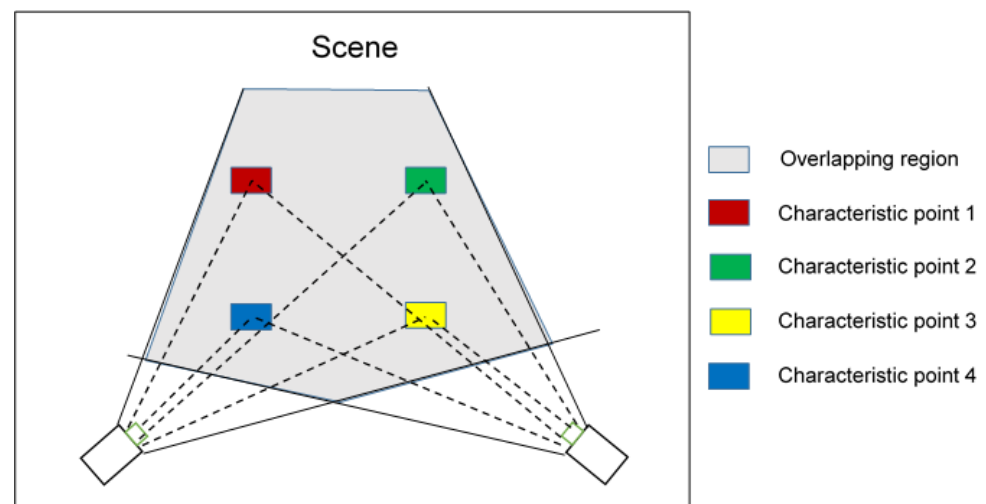

Figure 3. Active marks—points of interest. Colorful blocks represent the marks shown in Figure 2.

The centroid of each mark indicates the four reference coordinates for the transformation from each of the spaces generated by each camera to the global space, which is physically defined in the real world by measuring from one point to another, thus establishing the coordinates $(0,0)$ at the centroid of the blue rectangle. The other physical measurements are shown through an image in Figure 4; each physical measurement is expressed in centimeters.

A segmentation is carried out within the images obtained in each perspective to obtain the information used in the process of applying homography. Methods that are widely used in the literature were applied, such as morphological filters, area filters, and the detection of the central point of the objects of interest [64] (rectangles: red, yellow, green, and blue). Figure 5 shows the segmentation of the spaces of interest that are compared with the real coordinates.
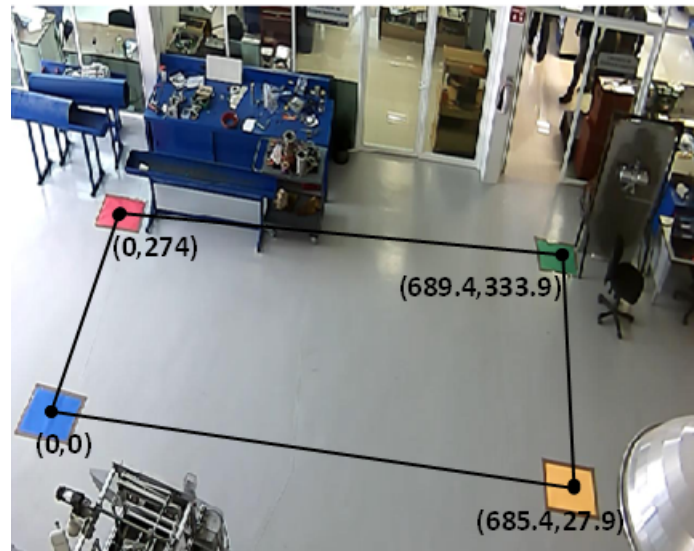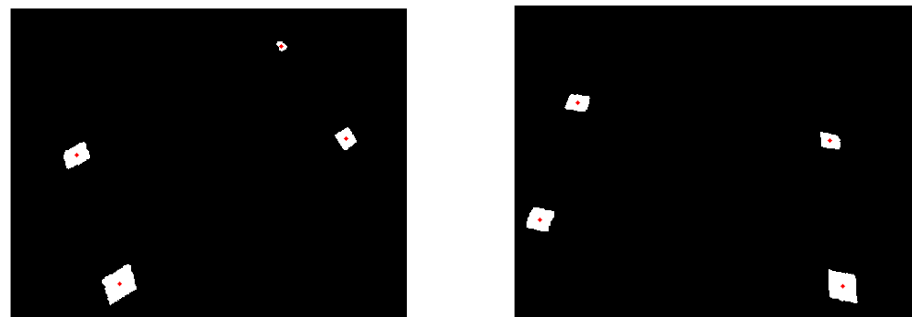

Figure 4. Original coordinates of references.

(**a**) Segmentation image from Camera 1.   (**b**) Segmentation image from Camera 2.

Figure 5. Segmentation of the perspective images. In the binary images, the white color represents the colorful block (Figure 2), and the red dots represent the centroid of each block.

The parameters defined up to this point allow for the application of the projective transformation for the generation of the global space where the trajectories found can be mapped using different methods for each camera. Trajectories, which are expressed in pixels, were obtained and, when mapping the global space, they become expressed in centimeters, thus defining the equivalence from one two-dimensional space to the other and leaving the trajectories in the global space, which are represented in Figure 6.

Figure 6 shows the 60 trajectories used that were found in the videos. The scale goes on the "x" axis from approximately -200 cm to 1200 cm and on the "y" axis from -200 cm to around 800 cm. To be able to provide feedback through augmented reality, a 3D space is required; for this reason, and due to the equivalences on the "x" and "y" plane, the "z" axis was defined with a height constant of 100 cm. This is because what is analyzed in the present work is the dynamics generated by objects in motion on the planes of "x" and "y".
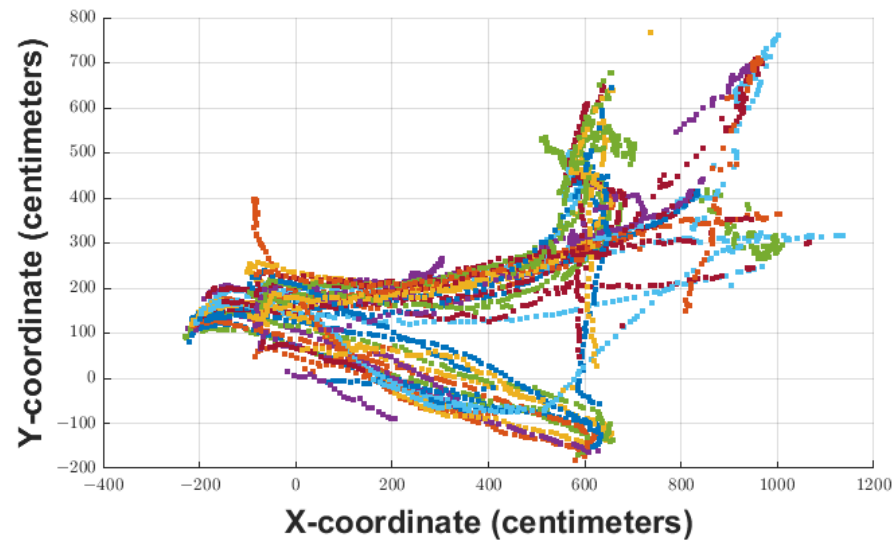
Figure 6. The trajectories in global space. The colored dots represent the different real trajectories generated by the objects of interest mapped to the global space.

3. Querying Trajectories

Query information is a process in which we seek to know the similarity between a reference and a data repository that allows us to simplify analysis in search of patterns or behaviors of interest. The process for performing the trajectory query depends on three essential factors: the trajectory that serves as a reference, the data repository that contains historical information, and a similarity measure.

In this work, to carry out queries, the generation of reference trajectories was compared with a data repository to determine the most similar ones and to be able to visualize them.

The generation of the reference trajectories was carried out from an augmented reality application in the global space. In this sense, a user generates a temporal trajectory from the application installed on a device (in this case, a tablet gadget); however, it is not exclusive to that device since devices with API 5.0 of the Android operating system or higher can run the application.

The application was developed in Unity3D, where all the trajectories of the moving objects were mapped to global space. Figure 7 shows, in a general way, how the application works. First, the reference trajectory was created in global space, as shown in Figure 8, and then the trajectory information was sent to a central unit that contains the data repository of the homogenized trajectories. In the same main unit, all the processing was carried out to calculate the similarity between the trajectories and to send the resulting information back to the application for viewing in the space where the information was acquired, that is, a client–server model was generated to send the data from the application to the central unit in both directions.

It is necessary to take into account the different aspects of carrying out querying and displaying the results through augmented reality, the first of which is that all the information obtained is in a known space where the information can be recovered. This allows for the creation of temporal reference information and a metric through which to estimate the similarity between the information.

By generating a global space where homogenized trajectories are represented, it is possible to propose a new metric to obtain similarities. For this, you can define a three-tuple as follows:

$$R = \{GlobalSpace, Reference, Metric\},$$

where *GlobalSpace* contains all the trajectories, in real coordinates, and is expressed in centimeters. *Reference* is the reference trajectory created from the global space generated. This is temporary information that is used for comparison with all the trajectories stored

in the data repository with those that define the similarity. *Metric* is the metric applied to order the information contained in the data repository in order to visualize the results that are more similar to the reference. Finally, *R* is a matrix that includes the trajectories sorted to start from the reference and the applied metric.
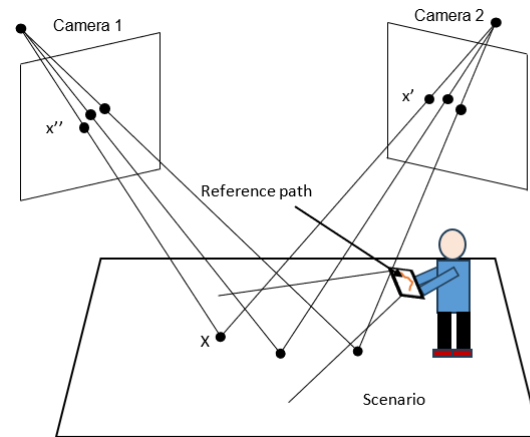

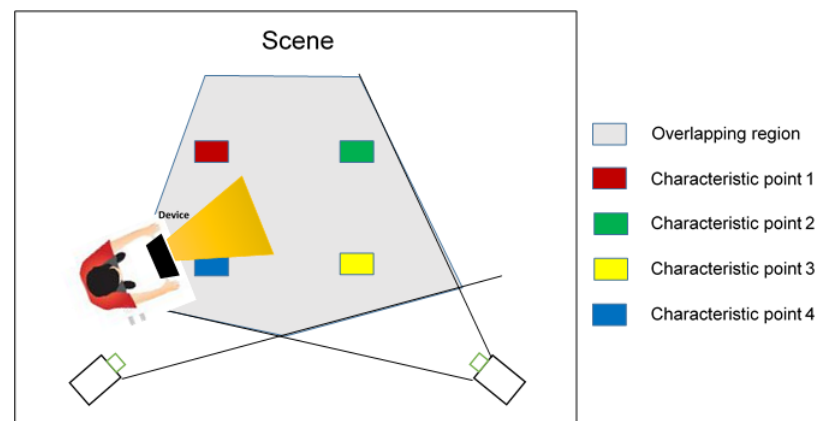**Figure 7.** Reference trajectory creation process—results display.

Figure 8. Reference trajectories.

4. Similarity Analysis

The most common way to define the similarity between trajectories is by applying some metric that allows for knowing the spatial or space–time distance [65], which exists between two trajectories $d(T_1, T_2)$.

This work aims to provide a proposal to unify and consult heterogeneous data, to generate and visualize information through interactive technologies, and to propose a novel way through which to determine the similarity between trajectories by analyzing their dynamics through a metric.

4.1. Metric for the Comparison of Movement Dynamics

The information supplied by a perspective camera is given in a two-dimensional space $\mathbb{R}^2$, where a vector space expressed by V has elements x, y, and ϵV, in such a way so that, for each trajectory generated by a moving object, there are pairs of points that evolve over time. When taking only one path, it can be represented as $T = \{x, y, t\}$, where x and y are the elements belonging to the vector space V, which has its domain in $\mathbb{R}^2$ and t is the time. Starting from this idea, it can be said that, in vector space, two orthogonal components can be parameterized and expressed as a function of time to define a function F.

Each of the trajectories stored in the data repository represents points in the two-dimensional space that can be characterized with a small set of parameters, that is, they can be defined by a representative function, with which the analysis of the trajectories can be performed on the functions and not on the total data that make up each of the trajectories. For this purpose, the discrete points of a trajectory can be modeled with a continuous parametric set of functions $F(t) = < f_1(t), \ldots, f_m(t) >$ with t parameters and m independent components of the trajectory.

The proposed similarity measure analyzes the difference between two different dynamics $F(t)$ and $G(t)$, and it involves calculating each f_i parametric for the m components using all of the samples of each trajectory, thereby obtaining, for each component i, an error as follows:

$$e_i = \sum_{j=1}^{m} |f_j^i - g_j^i|. \tag{2}$$

Given a vector of errors $e = [e_1, \ldots, e_n]$ under a norm of $||\mathbf{e}||$ represents the similarity between the trajectories, where $|| \cdot ||$ is any valid norm. Additionally, the proposed measure of similarity can calculate the similarity between the intervals defined as the vectors d and e of dimension $\times n$, with these being determined by the lower limit and upper limit between all components f_i. However, the expression $|f_j^i - g_j^i|$ strictly compares the dynamics, and it is totally dependent on the initial conditions of F and G. For this reason, it was observed that the behavior of the dynamics can be estimated by ∇F, which, in vector form, can be expressed by $\nabla F = < f_1', \ldots, f_n >$.

Then, the similarity between the system dynamics is denoted by e in the interval $t \in (d, e)$ as

$$e_i(f_i, g_i; d, e) = \int_d^e f_i'(t) - g_i'(t) \cdot dt, \tag{3}$$

and, consequently, the dissimilarity of the system is denoted by

$$d(F, G) = || < e_1(f_1; d_1, e_1), \ldots, e_1(f_n; d_n, e_n) > ||, \tag{4}$$

where F and G are the dynamics of a function. Since Equation (3) assumes that $f_i'(t) > g_i'(t)$, it can be rewritten as

$$M_{ij}(f_i, g_i; d, e) = \max\{e_i(f_i, g_i; d, e), e_i(f_j, g_i; d, e)\},$$
$$m_{ij}(f_i, g_i; d, e) = \min\{e_i(f_i, g_i; d, e), e_i(f_j, g_i; d, e)\}. \tag{5}$$

If the functions f_i and g_i have intersections, then it is possible to define $C = \{c_1, c_2, \ldots, c_h\}$ as the ordered intersection of points from the lowest to highest, where $\forall_{c_k \in C \wedge c_k \in (d,e)}$ $(f_i(c_k) - g_i(c_k)) = 0$. In this way, Equation (3) can be expressed in terms of XC as follows:

$$d_{ij}(f_i, g_i; d, e) = (M_{ij}(f_i, g_i; d, xc_1) - m_{ij}(f_i, g_i; d, xc_1)) + \ldots$$
$$= +(M_{ij}(f_i, g_i; xc_h, e) - m_{ij}(f_i, g_i; xc_h, e), \tag{6}$$

where Equation (6) represents the measure of similarity between f_i and g_i in the interval (d, e), thus expressing Equation (3) in a parametric way as follows:

$$\int_d^e ||(f_i'(t) - f_j'(t))||ds, \tag{7}$$

and representing each f_k as the representative points of each trajectory as follows:

$$\mathbf{f}_k = [f_{i1}, f_{i2}, \ldots, f_{im}]^T. \tag{8}$$

For more details, check Appendix A.

4.2. The Proposed Metric and Its Relationship with the Metric L_1

This section shows how the proposal in specific circumstances becomes equivalent to the L_1 metric. This becomes important because, in several scenarios, it will improve and simplify its computation, speeding up the process of trajectory querying. The scenario in which this becomes equivalent is whenever the similarity criteria are approximated as a linear polynomial.

The proposed metric can be seen as a composition of the metric L_1, but it produces different results in some aspects. Establishing whether two metrics are equivalent makes sense if the generated topological space has the same shape. The main difference between the two metrics is the domain and how the data are taken for comparison.

The metric L_1 takes the data in domain $\mathbb{R}^m$, subtracts for all components, and takes the absolute value. The proposed metric has its domain in $\mathbb{P}^m$, but the difference is the sense of instantaneous change, which coincides with the differences in the behavior of the trajectories. In this sense, metric L_1 can be seen as a particular case of the proposed metric when a first-order polynomial function represents the trajectories.

The proof is as follows: in having two functions $f_i(x)$, $f_j(x) \in \mathbb{P}^1$ with the form $f_i(x) = a_i x + c$; $f_j(x) = a_j x + c$ in the interval $(0,1) \in \mathbb{R}$, suppose $f_i(x) > f_j(x)$. The degree of similarity of these functions is calculated by

$$
\begin{aligned}
dif_{f_i f_j}(0,1) &= (M_{f_i f_j}(0,1) - m_{f_i f_j}(0,1)) \\
&= max(I(f_i,0,1), I(f_j,0,1)) - min(I(f_i,0,1), I(f_j,0,1)) \\
&= max(a_i, a_j) - min(a_i, a_j).
\end{aligned}
\tag{9}
$$

The second step is true because the derivative of any polynomial of degree 1 is the coefficient of the first-order term, and its area over the interval is the same coefficient. In the third step, we have $max(a,b) - min(a,b) = \| a - b \|$, where $\| \|$ is the absolute value. We can rewrite the expression as $dif_{f_i f_j}(0,1) = \| a_i - a_j \|$. Then, the proposed metric presented in a polynomial of degree 1 is the same as metric L_1, which we wanted to prove as being a particular case.

5. Proposed Metric Application

The proposed similarity measure is suitable for analyzing the trajectories of moving objects found in a scenario from which information is acquired through one or more sources of information (cameras). In these scenarios, the curves are smooth and can be well-fitted by a low-order polynomial. Let us define a trajectory of a moving object as $g = \{p_1, p_2, \ldots, p_n\}$, where each p_n represents the centroid of the defined object as $p_n = (x_n, y_n, t_n^T$. Parametric curves can model this trajectory as follows:

$$
f(t) = [g_x(t), g_y(t)]^T,
\tag{10}
$$

where t represents the sampling. Hence, we take, as an example, defining a second-order polynomial in such a way that the representative parametric functions of the trajectory are as follows: $g_x(t) = a_0 + a_1 X + a_2 X^2$ y $r_x(t) = b_0 + b_1 X + b_2 X^2$. These were calculated using the least squares method, a simple and well-established method in the literature.

Suppose we have two trajectories, g and r, for which we want to calculate similarity using the proposed metric. If we take one of the components of each path on the same axis, we have g_x and r_x, which are defined by a second-order polynomial like $g_x(t) = a_0 + a_1 X + a_2 X^2$ and $r_x(t) = b_0 + b_1 X + b_2 X^2$.

The proposed metric analyzes the behavior of the data and is invariant to the starting point of the trajectories. For this reason, the derivative of the functions is calculated as $g_x(t) = a_0 + a_1 X + a_2 X^2$ y $r_x(t) = b_0 + b_1 X + b_2 X^2$. Both curves intersect at $xc = \frac{a_0 - b_0}{2(a_1 - b_1)}$ for all $a_1 \neq b_1$. If $xc \in (d,e)$, then Equation (6) can be rewritten as

$$
\begin{aligned}
dif_{g_x, r_x}(d,e) &= (M_{g_x, r_x}(d, xc) - m_{g_x, r_x}(d, xc)) + \\
&\quad (M_{g_x, r_x}(xc, e) - m_{g_x, r_x}(xc, e)).
\end{aligned}
\tag{11}
$$

Figure 9 graphically shows the calculation performed using the proposed metric for a component of two different trajectories, where the shaded area is the area to be calculated.

The same process is repeated for each component of the trajectories, thus leaving the similarity defined by $\| D \|$, where D can be expressed for two components as

$$D = \begin{bmatrix} d_{g_x,r_x} \\ d_{g_y,r_y} \end{bmatrix}. \tag{12}$$

The metric expression has been reduced to a simple expression with parameters f_i, f_j, d, and e, which are functions to be compared and the interval limits in which they coincide. The computational complexity of the last expression is in the order of $O(k)$, where k is a constant.

Figure 9. Representation graphics of the proposed metric.

6. Augmented Reality for Querying and Visualization Data

Augmented reality systems have shown a novel way through which to visualize information, allowing for direct interaction between the real world and virtual elements that can be added to enrich the information contained in a scenario. The importance of using feedback through augmented reality in this work consists of generating queries and visualizing information in real-time to obtain a different analysis and interpretation by being present directly in the scenario where the trajectories of the objects of interest were raised.

In this work, the results obtained in the queryings were displayed via augmented reality through a client–server model. This model consists of a device (client) that helps to generate reference trajectories in the scenario where information on movement dynamics was previously acquired; these reference trajectories are sent and processed in a central unit (server) that returns the results for display to the device.

An application developed in Unity3D was installed on the device, which is designed to work in the global space that is generated where the trajectories of moving objects are mapped. Figure 10 shows roughly how the application works. First, the reference trajectory was created in the unified space, as shown in Figure 11, and then the trajectory information was sent to a central unit (server) that contains the trajectory data repository. In the same central unit, all the processing was carried out to calculate the similarities between the trajectories stored in the data repository, and the resulting information was then sent back to the application for viewing.
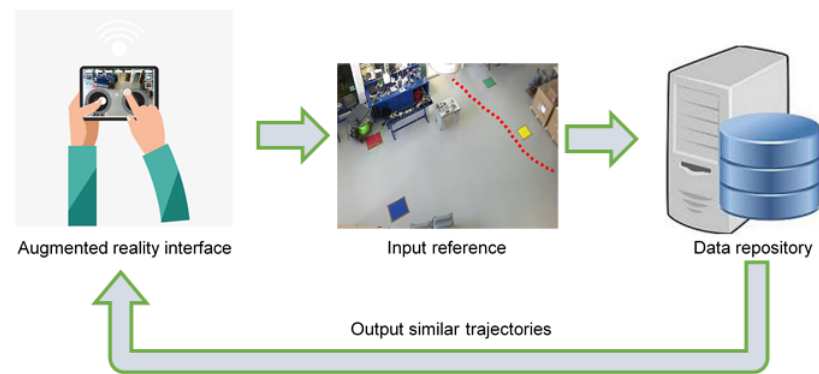
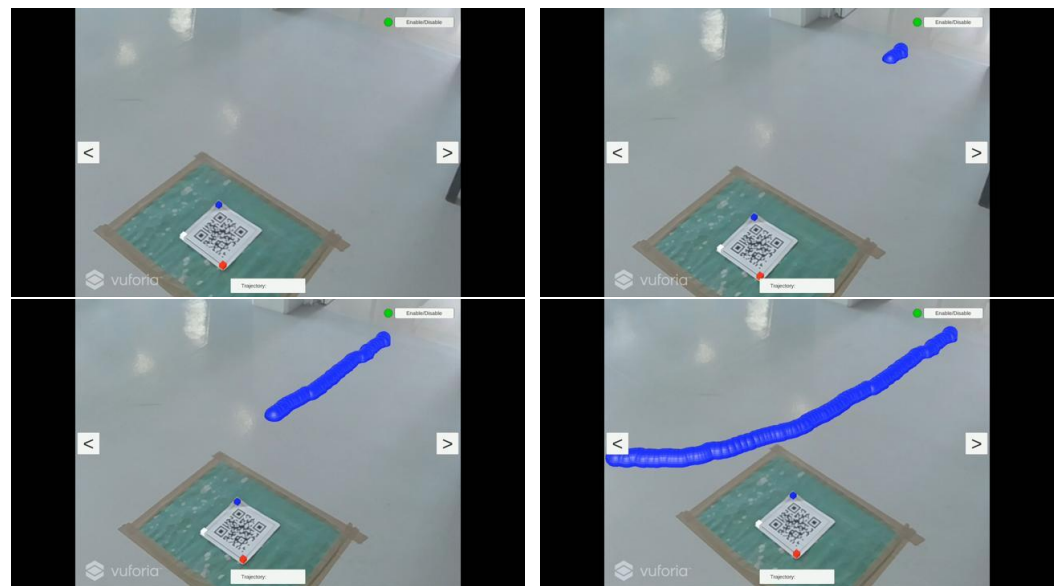**Figure 10.** Breakdown of the creation of the reference trajectory—result visualization.

Figure 11. Process of creation of the reference trajectory—result visualization. Blue dots represent the construction of the reference trajectory created by the user. The QR code is the active mark necessary for the augmented reality application.

The communication between the developed application and the central unit was carried out by employing the TCP/IP protocol. The reference trajectory was encoded in a string message that traveled to the central unit; it was decoded for use in comparison with the trajectories stored in the data repository, and the ordering was performed according to the metric used. The five most similar trajectories to the reference were selected, encoded again in a message type *string*, then they traveled back to the application where they were decoded again for use, and then the results were displayed in augmented reality, as seen in Figure 12a,b.

For the performance of the augmented reality application, the following points can be commented on:

1. Communication: The complete system is provided by Ethernet communication. The video cameras are Ethernet interfaces. The communication is achieved via a 1 Mbps camera streaming. The Ethernet infrastructure becomes a 1 Gb Ethernet VPN. Simultaneously, we process two cameras, which, in peak situations, give around 20 Mbs sustained in continuous streaming.
2. Computation of data for storage: The backend server aims to execute the tracking algorithm and store it in the global reference space. The transformation in a global reference space becomes complex because the cameras become fixed, and the projection is based on the active markers in the scenario. Once the camera computes the matrix projection, it becomes fixed until the camera position or orientation changes.

3. Computation of data for querying: This process involves the client (personal gadget) and requires sampling in a field of view of the orientation into the scene. A client app captures the trajectory for querying after locating the active markers. Projection markers are commutated to transform into active markers and global reference space. This operation is performed once after capturing the trajectory to be consulted as a reference.

4. Displaying process: This represents the most intensive resource because the position and orientation are continuously checked; the current camera orientation is continuously projected onto the trajectory into the reference space that communicates similar information to the server.

5. The displaying process represents a 3D view that uses the client's current GPU or graphic interface. This operation becomes limited by the client and trajectory complexity dynamics expressed in the device. On the other hand, the backend server computes the similarity of the trajectories and sends the point trajectories to be shown on the device.

(**a**)

(**b**)

Figure 12. (**a**) Server: communication process. (**b**) Client: communication process.

7. Results

The work was designed for application to any metric that can compare vectors of different lengths, such as those found in [27,66–70]. To prove the validity of the proposed metric, tests were first carried out with the trajectories being generated through a code developed in MATLAB; the trajectories were mapped to the global space such that the coordinates could be expressed in millimeters. The objective was to make a comparison with a widely used metric to compare data of different lengths, such as the Hausdorff distance [71]. Figure 13 shows the two trajectories to be compared.

The process was carried out by applying the proposed metric where an orthogonal analysis was carried out for each trajectory, as well as by dividing the information into its components, "x" and "y". Subsequently, the derivative was applied, as shown in Figure 14. An adjustment of them was calculated by applying the least squares method, and Figure 15 summarizes the abovementioned process. Finally, the proposed metric and the Hausdorff distance were applied, and a notable difference in the results was obtained.

Figure 16 shows the two trajectories to be compared. The third figure has two equal trajectories only out of phase in space, so it can show how the proposed metric works since it analyzes the dynamics of the data, which is why the value obtained should be equal to or very close to zero. Table 3 shows the results of each of the tests carried out by applying the same metrics.
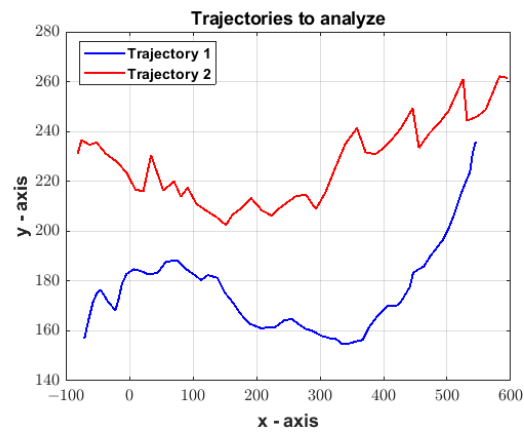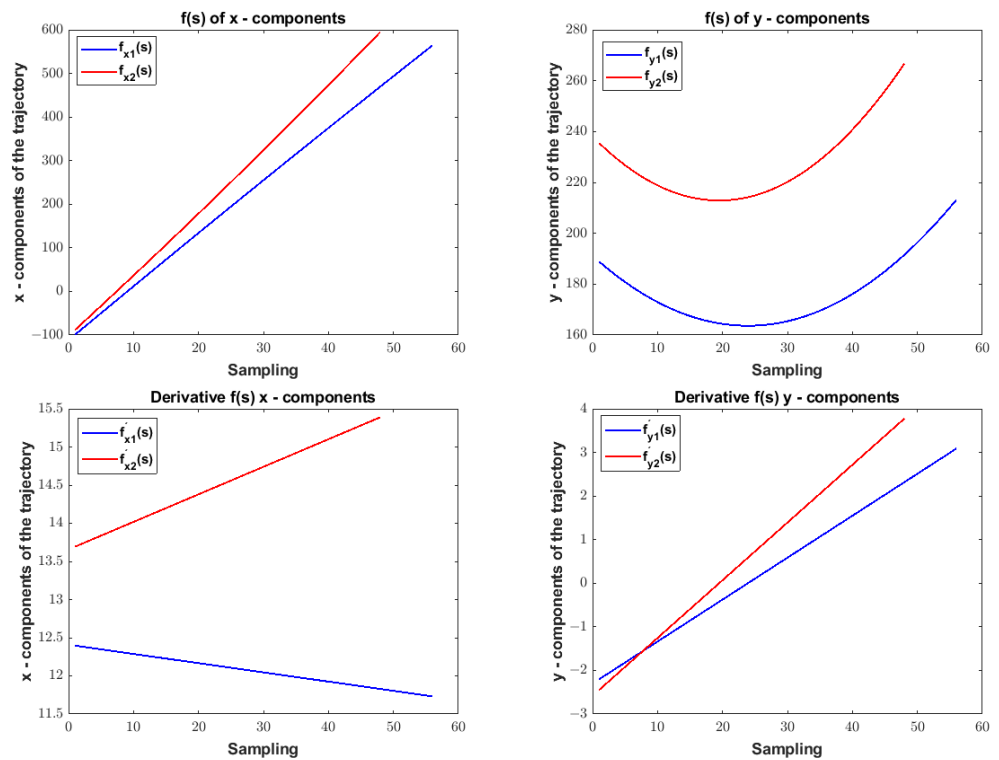
Figure 13. Trajectories to compare.
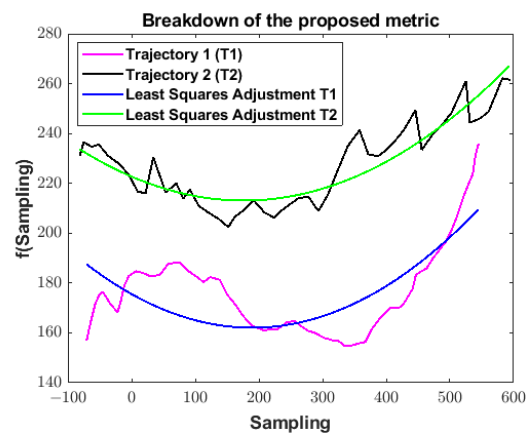

Figure 14. Process of the proposed metric.
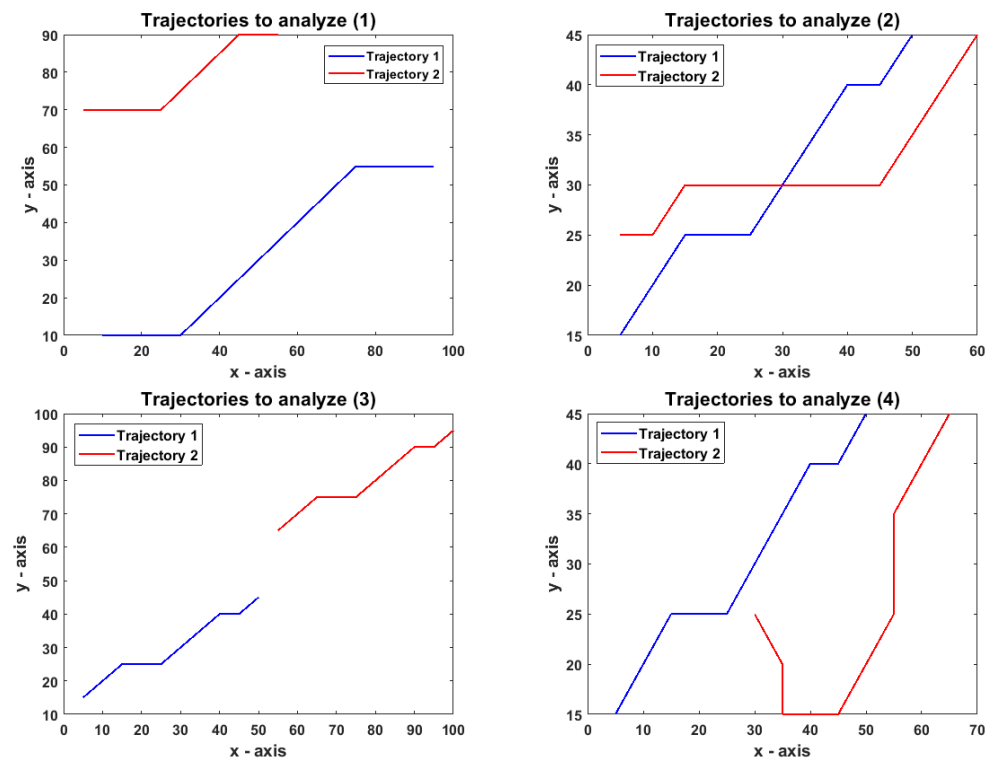

Figure 15. Breakdown of the proposed metric.

Figure 16. Example trajectories to compare.

Table 3. Results of the comparative proposed metric vs. the Hausdorff.

Trajectories	Proposed Metric	Hausdorff
Trajectories in Figure 13	118.12 units	8.80 units
Trajectories in Figure 16, first	10.00 units	20 units
Trajectories in Figure 16, second	19.14 units	0 units
Trajectories in Figure 16, third	5.6843×10^{-12} units	10 units
Trajectories in Figure 16, fourth	27.64 units	10 units

Once the validation of the proposed metric has been reviewed by comparing it with one of the metrics used in the literature, its implementation is carried out through real-time query generation that is applied augmented reality. The reference trajectory generation was carried out on the tablet gadget so that it could be sent to the central unit where a metric was applied and the results were displayed. Figure 11 shows an example of elaborating a reference trajectory in the augmented reality application.

A spatial metric tolerant to trajectories of different lengths was applied first to obtain results and to compare them with the proposed metric. As mentioned before, the Hausdorff distance metric was used. Figure 17 presents the results when applying this metric, and it shows a three-dimensional space that exemplifies the trajectories seen through augmented reality, where each one of them can be shown in an orderly manner starting with the one most similar to the reference followed by a less similar one (performed according to the similarity results obtained with the metric used). The user generates the blue color trajectory, which is the reference trajectory. Based on the results and the study of the Hausdorff distance, it can be seen that, with a spatial metric, the results are trajectories closer to the reference user generated.
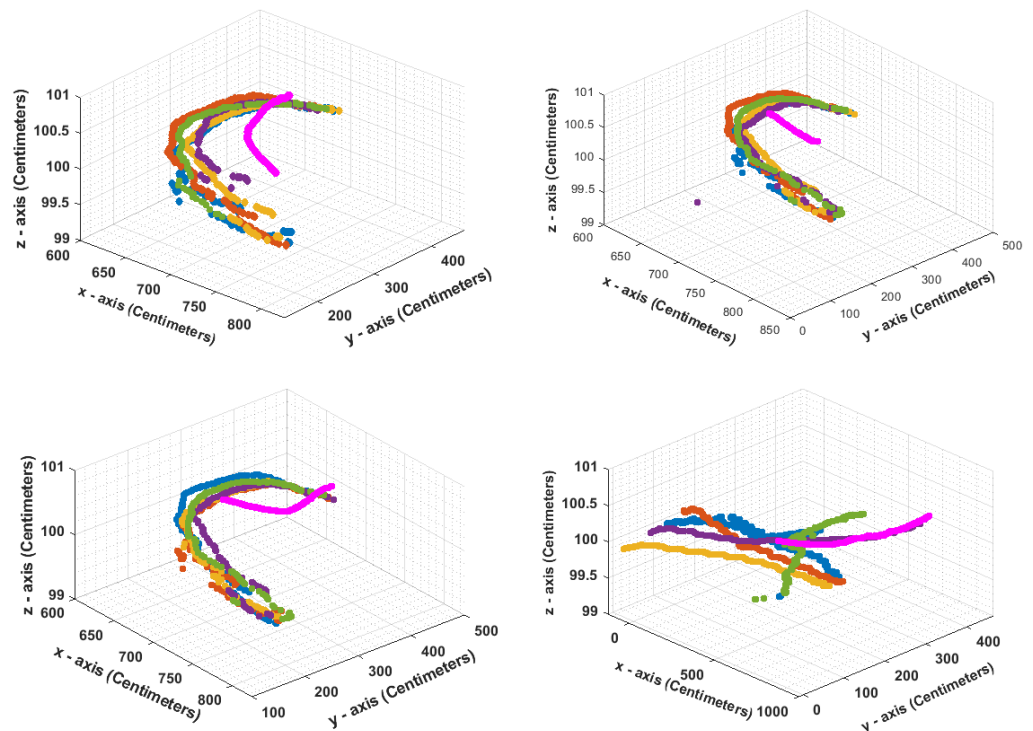
Figure 17. Graphical representation of the result with Hausdorff. The colorful dots represent the different trajectories most similar to the reference, where dots in magenta color are the reference trajectory generated by the user.

The proposed metric differs from the metrics that are currently better known and used in the literature because it analyzes the trajectory dynamics. By making use of the derivative of a polynomial of degree n, it can be assumed that the analysis is about the behavior of the trajectory, and that it positions the trajectories to be compared at a point of origin, thus eliminating the translation that exists in the difference between one trajectory and another. For this reason, it does not matter that, spatially, the distance between a trajectory similar to the reference is very far away in the plane.

When applying the proposed metric, Figure 18 shows the results of the first query seen from the tablet gadget, and it shows two different trajectories in red and blue; the trajectory in blue is the reference trajectory, while the trajectory in red is the most similar according to the applied metric. In Figure 19, the five trajectories most similar to the created reference are graphically shown; each of them can be viewed in the augmented reality application in an orderly manner. For practical purposes, it is displayed one by one; however, with some quick changes to the source code, more similar trajectories can be displayed simultaneously.

Each color represents a different trajectory, with the magenta color being the reference trajectory created from the augmented reality application. Figures 20 and 21 show the new query made and the results.

Figure 22 shows the results of the querying, and they were displayed in the real world through the augmented reality application generated for this work.

Figure 23 shows several of the queries that were made applying the proposed metric, where the magenta trajectory represents the trajectory generated through augmented reality. These results are shown in 3D graphs for practicality to show all of the points that represent the trajectories found in the data repository (which stores the trajectories captured by the objects of interest).

In addition to the results shown when applying the metrics, the topological space created when applying the proposed metric can be shown. In Section 5, it was explained that each trajectory is parameterized so that there are two orthogonal components to analyze. There is a representation that employs a parametric function on the x axis and

an expression by a parametric function on the *y* axis; when the parametric functions are analyzed independently, a valid rule was applied to obtain the result of the similarity according to the dynamics generated by the trajectories to be compared. Figure 24 shows the topological space generated from the reference trajectory when performing the query with the data repository.
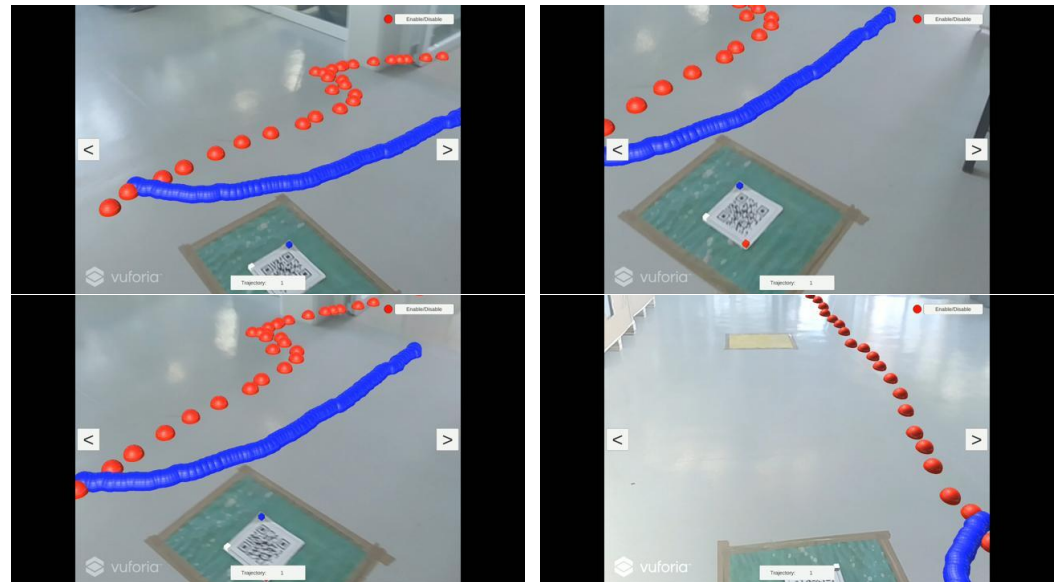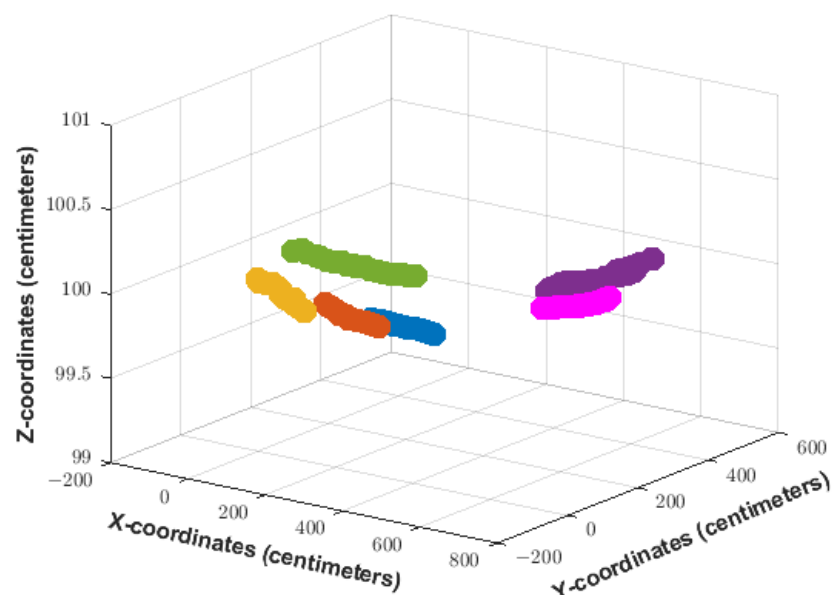

Figure 18. Results from applying the metric proposal. Blue dots represent the reference trajectory created by the user, and red dots represent a similar trajectory as from the reference. The QR code is the active mark necessary for the augmented reality application.

Figure 19. Graphical representation of the results (Query 1). The colorful dots represent the different trajectories most similar to the reference, where dots in magenta color are the reference trajectory generated by the user.
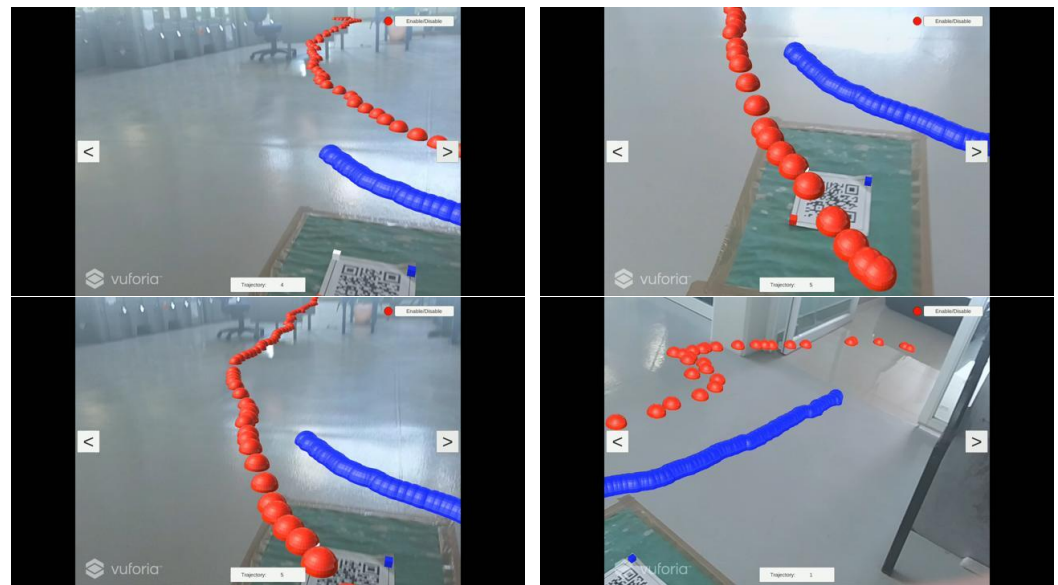
Figure 20. Results of applying the metric proposal and visualization through augmented reality. The blue dots represent the trajectory created by the user, and the red dots represent one of the most similar trajectories according to the applied metric. The QR code is the active mark necessary for the augmented reality application.
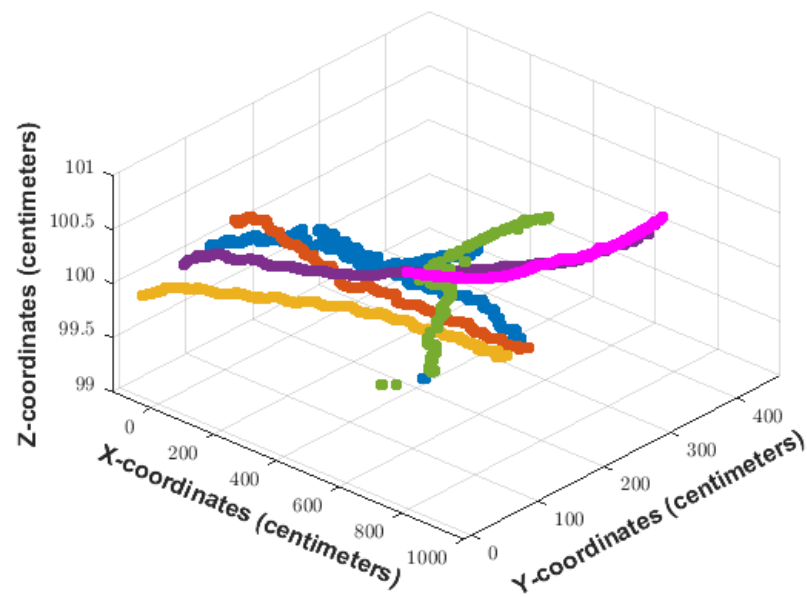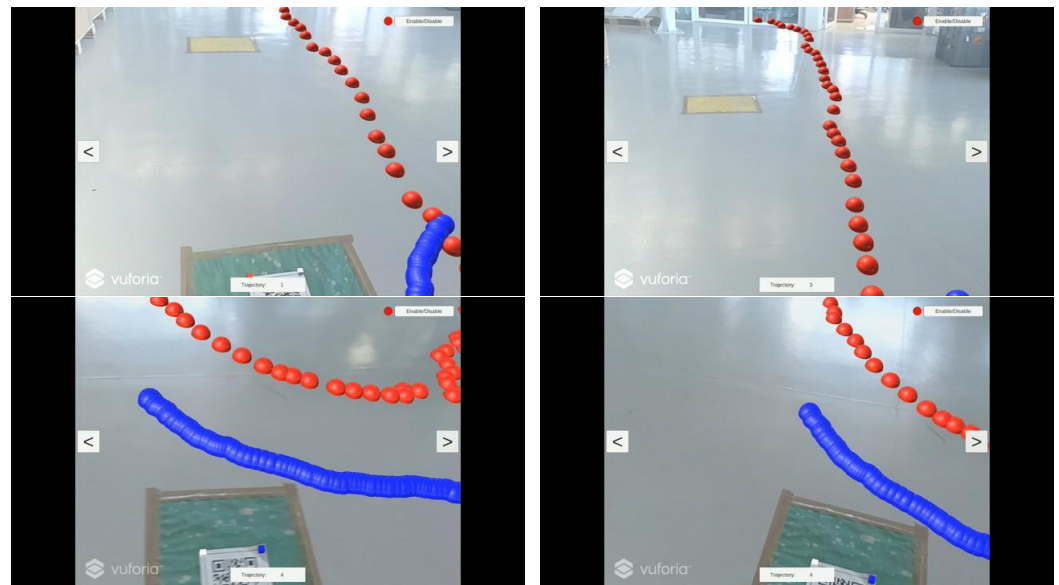

Figure 21. Graphical representation of the result (Query 2). The colorful dots represent the different trajectories most similar to the reference, where dots in magenta color are the reference trajectory generated by the user.

Figure 22. The Figure shows new results by applying the metric proposal and visualization through augmented reality. The blue dots represent the trajectory created by the user, and the red dots represent one of the most similar trajectories according to the applied metric. The QR code is the active mark necessary for the augmented reality application.
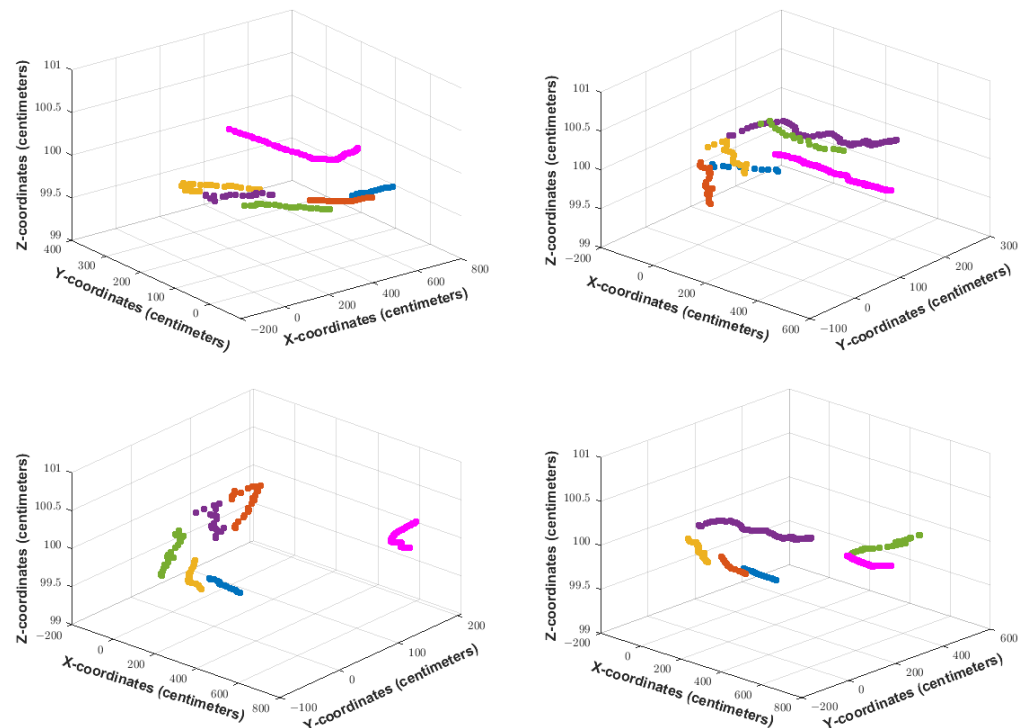

Figure 23. Graphical representation of the results. The colorful dots represent the different trajectories most similar to the reference, where dots in magenta color are the reference trajectory generated by the user.
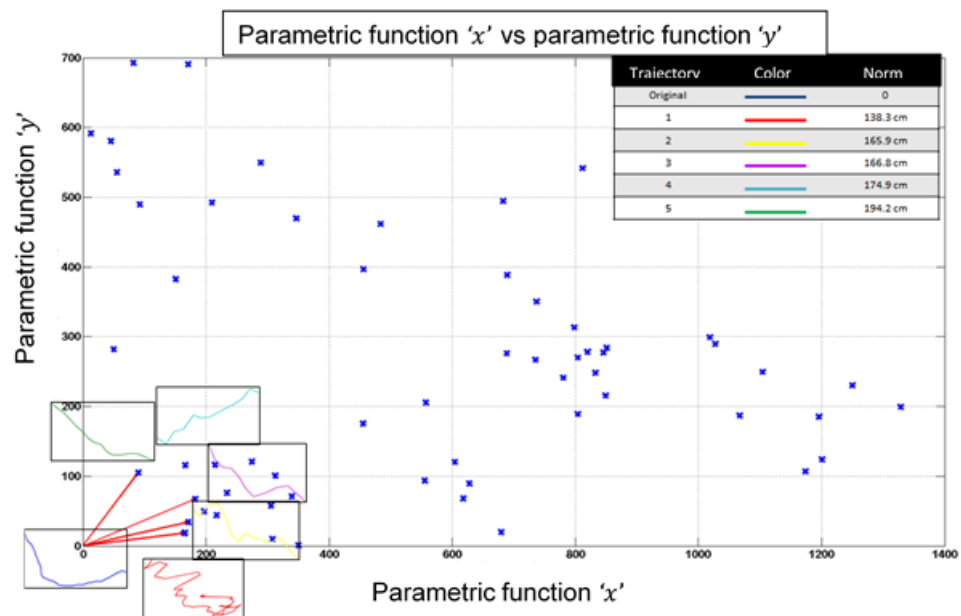
Figure 24. The topological space of the proposed metric. The blue dots represent different trajectories ordered according to the proposed metric, where the dispersion is obtained when a dynamic analysis is realized.

It also follows that, regardless of the scenario to be analyzed, the number of dynamically different trajectories tends to be limited and generates classes. For the trajectories stored in the data repository and the reference trajectory used, Figure 25 shows the number of classes that can be obtained with this information.
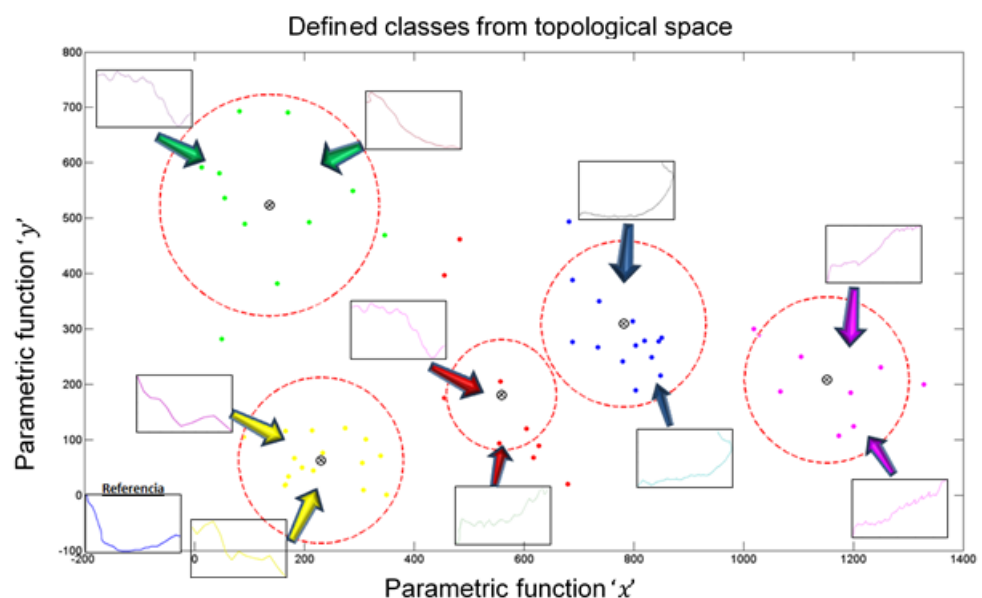

Figure 25. Generated classes from the reference. The colorful circles represent different classes obtained from applying the proposed metric and analyzing the dynamic of trajectories.

8. Discussion

Multi-camera systems have proven to be versatile and passive tools for detecting information of interest; however, the amount of information acquired has grown exponentially and has made its analysis a significant challenge when determining patterns and behavior within the monitored scenarios. The techniques and models used for analysis are increasingly new but simultaneously more complex because they require the help of

advanced technology for their application and visualization, and this is in addition to generating heterogeneous information that is difficult to compare or manipulate.

In the present work, a way in which to work with heterogeneous information was sought by mapping it onto a global space with the help of methods that are widely used in the literature, such as homography, as well as by proposing a new metric to compare said information (achieved by taking the dynamics as a reference for the movement of objects of interest (pedestrians)). This provides a metric that does not focus on the temporal or spatiotemporal analysis that exists in the literature. Likewise, an application was developed to create reference and visualization information within the vision space of a test camera system, with excellent results showing the real-time queries of trajectories similar to the generated reference.

As there were implementation issues, the proposal involves a central server, which stores and creates the global space that is projected in each camera's field-of-view motion data storage, as well as projects the current camera view of the query information made for a mobile device. The communication occurs over a VPN, where the server responds as a service for querying, and the communication is given by exchanging Jason files that provide the tuples of the trajectories required.

This approach elevates technologies to manage several information sources, and it represents all data in a common workspace once the common workspace is used as a global reference for which external queries become possible. In the proposal, queries become visual and are relative to the observer position projecting into the global space. One of the differences that was proposed when carrying out this work was the parameterization of trajectories generated by people within the test space that was built for the development of this research. In this case, they were parameterized using the least squares methodology, which represented the trajectories as square functions because, normally, the trajectories that are generated have smooth curves. In addition, analysis via orthogonal components was used to show that the metric applies to an n-dimensional space, and that it relates to the results at the end by calculating the magnitude of the orthogonal components obtained. As a result, the difference in dynamics between the trajectories was compared. This methodology is innovative as it applies the derivative of the function that is represented to the trajectories independently, thus eliminating spatiality and the integral as a means of defining the area between the trajectories, thereby allowing for an analysis of the data behavior. Additionally, the similarity function is useful because it provides a way through which to match the heterogeneous information stored in a global space.

Future work will include the following:

- The possibility of representing the movement dynamics with polynomials of a different order.
- The application of the arc length technique.
- The homogenization of information with passive marks to use information through augmented reality.
- The incorporation of more sources of heterogeneous information.
- Dynamically selecting markers to orientate the systems.

9. Conclusions

In artificial vision systems, many challenges must be addressed to achieve an information unification that is better for interpreting the events occurring in specific settings. In the case of distributed systems, one of the biggest challenges is the fusion of information from different sources (cameras), which is needed to compare heterogeneous information. In the present work, the authors tried to explain that having a distributed system can obtain better coverage of the area from which data can be acquired, thus increasing the complexity of the information analysis considerably. By proposing a way through which to generate a global space, mappings of different spaces can be made to compare heterogeneous information and develop a unified space that allows information to be fed back by applying a valid metric. Finally, the interactive visualization of data provides a different way of presenting

the results to improve their interpretation directly in the real world. The use of augmented reality as a means of generating and displaying information allows for, in a novel way, real-time analysis to be carried out directly in the space where pedestrians generate trajectories; this could be applied in video surveillance systems to compare the unusual trajectories related to criminal acts by showing the trajectory generated by a specific person.

Appendix A

To be a metric, Equation (6) must satisfy the following four properties:

1. $d(x, y) = 0 \Leftrightarrow x = y$, i.e., the distance from a point to itself is zero;
2. $d(x, y) \geq 0$ positivity;
3. $d(x, y) = d(y, x)$ symmetry;
4. $d(x, z) \leq d(x, y) + d(y, z)$ triangle inequality.

Demonstration of the properties with the proposed metric.

1. $d_{ii}(d, e) = 0$. According to Equation (5), the term on the left can be expressed as follows: $M_{ii}(d, e) - m_{ii}(d, e) = 0$. Expanding the terms, we have

$$max(I(f_i, d, e), I(f_i, d, e)) - min(I(f_i, d, e), I(f_i, d, e)) = 0. \tag{A1}$$

In general terms $I(f_i, d, e) = k$, so, by substituting it into Equation (A1), it follows that $max(k, k) = k$ y $min(k, k) = k$—in other words $k - k = 0$.

2. $d_{ij}(d, e) \geq 0$. Si $d_{ij}(d, e) = 0$. If $d_{ij}(d, e) = 0$, then, at that time, $d_{ii}(d, e) = 0$; otherwise, the first term on the left can be expressed as $M_{ij}(d, e) - m_{ij}(d, e) \geq 0$. Expanding the terms, we have

$$max(I(f_i, d, e), I(f_j, d, e)) - min(I(f_i, d, e), I(f_j, d, e)) \geq 0. \tag{A2}$$

Again, generally $I(f_i, d, e) = k$ y $I(f_j, d, e) = l$. Using Equation (A2), we have the following options:

- $k > 0$ y $l < 0$. Becomes $k > l$.
- $k > 0$ y $l > 0$. Then $max(k, l) - min(k, l) > 0$.
- $k < 0$ y $l < 0$. Becomes $max(k, l) - min(k, l) > 0$, because k y l are negatives.
- $k < 0$ y $l > 0$. Then $l > k$.

3. $d_{ij}(d, e) = d_{ji}(d, e)$. The left-hand term can be expanded to $d_{ij}(d, e) = M_{ij}(d, e) - m_{ij}(d, e)$; $d_{ji}(d, e) = M_{ji}(d, e) - m_{ji}(d, e)$. Applying the auxiliary function defined in Equation (5), we can rewrite this as follows:

$$d_{ij}(d, e) = max(I(f_i, d, e), I(f_j, d, e)) - min(I(f_i, d, e), I(f_j, d, e))$$
$$d_{ji}(d, e) = max(I(f_j, d, e), I(f_i, d, e)) - min(I(f_j, d, e), I(f_i, d, e)). \tag{A3}$$

Assuming that $I(f_i, d, e) = k$ y $I(f_j, d, e) = l$, then Equation (A3) can be represented as:

$$
\begin{aligned}
d_{ij}(d,e) &= max(k,l) - min(k,l) \\
d_{ji}(d,e) &= max(l,k) - min(l,k),
\end{aligned}
\tag{A4}
$$

so that $d_{ij}(d,e) = d_{ji}(d,e)$ can be expressed by

$$
max(k,l) - min(k,l) = max(l,k) - min(l,k).
\tag{A5}
$$

If $k > l$

$$
k - l = k - l.
\tag{A6}
$$

If $l > k$

$$
l - k = l - k.
\tag{A7}
$$

If $k = l$

$$
\begin{aligned}
k - l &= 0 \\
l - k &= 0 \\
&\therefore \\
0 &= 0.
\end{aligned}
\tag{A8}
$$

4. $d_{ik}(d,e) \le d_{ij}(d,e) + d_{jk}(d,e)$. The expression on the left can be rewritten as follows:
    $[max(I(f_i,d,e), I(f_k,d,e)) - min(I(f_i,d,e), I(f_k,d,e))] \le [max(I(f_i,d,e), I(f_j,d,e)) - min(I(f_i,d,e), I(f_j,d,e))] + [max(I(f_j,d,e), I(f_k,d,e)) - min(I(f_j,d,e), I(f_k,d,e))]$
    Assuming that $I(f_i,d,e) = A_i$, $I(f_j,d,e) = A_j$, $I(f_k,d,e) = A_k$, then only one of the following cases is possible:

    (a)　$A_i > A_j > A_k : A_i - A_k \le (A_i - A_j) + (A_j - A_k)$;
    (b)　$A_i > A_k > A_j : A_i - A_k \le (A_i - A_j) + (A_k - A_j)$;
    (c)　$A_k > A_i > A_j : A_k - A_i \le (A_i - A_j) + (A_k - A_j)$;
    (d)　$A_k > A_j > A_i : A_k - A_i \le (A_j - A_i) + (A_k - A_j)$;
    (e)　$A_j > A_i > A_k : A_i - A_k \le (A_j - A_i) + (A_j - A_k)$;
    (f)　$A_j > A_k > A_i : A_k - A_i \le (A_j - A_i) + (A_j - A_k)$.

    With these verifications, it was concluded that $d_{ij}(d,e)$ is a metric, which is what we wanted to demonstrate.

References

1. Elharrouss, O.; Almaadeed, N.; Al-Maadeed, S. A review of video surveillance systems. *J. Vis. Commun. Image Represent.* **2021**, *77*, 103116. [CrossRef]
2. Subudhi, B.N.; Rout, D.K.; Ghosh, A. Big data analytics for video surveillance. *Multimed. Tools Appl.* **2019**, *78*, 26129–26162. [CrossRef]
3. Olagoke, A.S.; Ibrahim, H.; Teoh, S.S. Literature survey on multi-camera system and its application. *IEEE Access* **2020**, *8*, 172892–172922. [CrossRef]
4. Villani, M.L.; De Nicola, A.; Bouma, H.; van Rooijen, A.; Räsänen, P.; Peltola, J.; Toivonen, S.; Guarneri, M.; Stifini, C.; De Dominicis, L. A Modular Architecture of Command-and-Control Software in Multi-Sensor Systems Devoted to Public Security. *Information* **2023**, *14*, 162. [CrossRef]
5. Mezhenin, A.; Izvozchikova, V.; Ivanova, V. Use of Point Clouds for Video Surveillance System Cover Zone Imitation. In *CEUR Workshop Proceedings*; CEUR-WS: Aachen, Germany, 2019; Volume 2344. Available online: https://ceur-ws.org/Vol-2344/short3.pdf (accessed on 7 December 2023).
6. Jain, S.; Ananthanarayanan, G.; Jiang, J.; Shu, Y.; Gonzalez, J. Scaling video analytics systems to large camera deployments. In Proceedings of the 20th International Workshop on Mobile Computing Systems and Applications, Santa Cruz, CA, USA, 27–28 February 2019; pp. 9–14.
7. Qiu, S.; Zhao, H.; Jiang, N.; Wang, Z.; Liu, L.; An, Y.; Zhao, H.; Miao, X.; Liu, R.; Fortino, G. Multi-sensor information fusion based on machine learning for real applications in human activity recognition: State-of-the-art and research challenges. *Inf. Fusion* **2022**, *80*, 241–265. [CrossRef]
8. Kwon, O.; Lee, N.; Shin, B. Data quality management, data usage experience and acquisition intention of big data analytics. *Int. J. Inf. Manag.* **2014**, *34*, 387–394. [CrossRef]
9. Yadav, C.; Wang, S.; Kumar, M. Algorithm and approaches to handle large Data—A Survey. *arXiv* **2013**, arXiv:1307.5437.

10. IntelliVision Now Inside 4 Million Smart Cameras—Leader in AI-Based Video Analytics Software. Available online: https://www.intelli-vision.com/news/intellivision-now-inside-4-million-smart-cameras-leaderin-ai-based-video-analytics-software/ (accessed on 10 November 2018).

11. Truong, H.L.; Fahringer, T. Self-managing sensor-based middleware for performance monitoring and data integration in grids. In Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium, Denver, CO, USA, 4–8 April 2005; pp. 156–163. [CrossRef]

12. Khalid, Z.; Fisal, N.; Rozaini, M. A survey of middleware for sensor and network virtualization. *Sensors* **2014**, *14*, 24046–24097. [CrossRef]

13. Benayache, A.; Bilami, A.; Barkat, S.; Lorenz, P.; Taleb, H. MsM: A microservice middleware for smart WSN-based IoT application. *J. Netw. Comput. Appl.* **2019**, *144*, 138–154. [CrossRef]

14. Eleftherakis, G.; Pappas, D.; Lagkas, T.; Rousis, K.; Paunovski, O. Architecting the IoT paradigm: A middleware for autonomous distributed sensor networks. *Int. J. Distrib. Sens. Netw.* **2015**, *11*, 139735. [CrossRef]

15. Akanbi, A.K.; Masinde, M. Semantic interoperability middleware architecture for heterogeneous environmental data sources. In Proceedings of the 2018 IST-Africa Week Conference (IST-Africa), Gaborone, Botswana, 9–11 May 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–10.

16. García-Huerta, J.M.; Jiménez-Hernández, H.; Herrera-Navarro, A.M.; Hernández-Díaz, T.; Terol-Villalobos, I. Modelling dynamics with context-free grammars. In *Video Surveillance and Transportation Imaging Applications*; International Society for Optics and Photonics: San Francisco, CA, USA, 2014; Volume 9026, p. 902611.

17. Rezaee, K.; Rezakhani, S.M.; Khosravi, M.R.; Moghimi, M.K. A survey on deep learning-based real-time crowd anomaly detection for secure distributed video surveillance. *Pers. Ubiquitous Comput.* **2021**, *27*, 1–17. [CrossRef]

18. Bouma, H.; Villani, M.L.; van Rooijen, A.; Räsänen, P.; Peltola, J.; Toivonen, S.; De Nicola, A.; Guarneri, M.; Stifini, C.; De Dominicis, L. An integrated fusion engine for early threat detection demonstrated in public-space trials. *Sensors* **2023**, *23*, 440. [CrossRef] [PubMed]

19. Wang, W.; Xie, Y.; Tang, L. Hierarchical Clustering Algorithm for Multi-Camera Vehicle Trajectories Based on Spatio-Temporal Grouping under Intelligent Transportation and Smart City. *Sensors* **2023**, *23*, 6909. [CrossRef]

20. Li, X.; Hu, W.; Shen, C.; Zhang, Z.; Dick, A.; Hengel, A.V.D. A survey of appearance models in visual object tracking. *ACM Trans. Intell. Syst. Technol. (TIST)* **2013**, *4*, 58. [CrossRef]

21. Malik, N.U.R.; Abu-Bakar, S.A.R.; Sheikh, U.U.; Channa, A.; Popescu, N. Cascading Pose Features with CNN-LSTM for Multiview Human Action Recognition. *Signals* **2023**, *4*, 40–55. [CrossRef]

22. Joshi, K.A.; Thakore, D.G. A survey on moving object detection and tracking in video surveillance system. *Int. J. Soft Comput. Eng.* **2012**, *2*, 44–48.

23. Adachi, T.; Hayashi, N.; Takai, S. Cooperative target tracking by multiagent camera sensor networks via Gaussian process. *IEEE Access* **2022**, *10*, 71717–71727. [CrossRef]

24. Li, Y.; Yang, G.; Su, Z.; Li, S.; Wang, Y. Human activity recognition based on multienvironment sensor data. *Inf. Fusion* **2023**, *91*, 47–63. [CrossRef]

25. Chen, K.-Y.; Chou, L.-W.; Lee, H.-M.; Young, S.-T.; Lin, C.-H.; Zhou, Y.-S.; Tang, S.-T.; Lai, Y.-H. Human motion tracking using 3D image features with a long short-term memory mechanism model—An example of forward reaching. *Sensors* **2022**, *22*, 292. [CrossRef]

26. Shidik, G.F.; Noersasongko, E.; Nugraha, A.; Andono, P.N.; Jumanto, J.; Kusuma, E.J. A systematic review of intelligence video surveillance: Trends, techniques, frameworks, and datasets. *IEEE Access* **2019**, *7*, 170457–170473. [CrossRef]

27. Yang, N.; Zheng, J.; Liu, Q.; Cao, Y. A Novel Trajectory Similarity Evaluation Method in VANETs. *Int. J. Multimed. Ubiquitous Eng.* **2014**, *9*, 183–192. [CrossRef]

28. Jung, S.; Lee, Y.S.; Lee, Y.; Lee, K. 3D reconstruction using 3D registration-based ToF-stereo fusion. *Sensors* **2022**, *22*, 8369. [CrossRef] [PubMed]

29. Kang, Z.; Yang, J.; Yang, Z.; Cheng, S. A review of techniques for 3d reconstruction of indoor environments. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 330. [CrossRef]

30. Aharchi, M.; Ait Kbir, M. A review on 3D reconstruction techniques from 2D images. In *Innovations in Smart Cities Applications Edition 3: The Proceedings of the 4th International Conference on Smart City Applications*; Springer International Publishing: Cham, Switzerland, 2020; Volume 4, pp. 510–522.

31. Yilmaz, A.; Javed, O.; Shah, M. Object tracking: A survey. *ACM Comput. Surv. (CSUR)* **2006**, *38*, 13. [CrossRef]

32. Jiménez-Hernández, H.; González-Barbosa, J.J.; García-Ramírez, T. Detecting abnormal vehicular dynamic at intersection based on an unsupervised learning approach and a stochastic model. *Sensors* **2010**, *10*, 7576–7601. [CrossRef]

33. Parekh, H.S.; Thakore, D.G.; Jaliya, U.K. A survey on object detection and tracking methods. *Int. J. Innov. Res. Comput. Commun. Eng.* **2014**, *2*, 2970–2978.

34. Singla, N. Motion detection based on frame difference method. *Int. J. Inf. Comput. Technol.* **2014**, *4*, 1559–1565.

35. Fortun, D.; Bouthemy, P.; Kervrann, C. Optical flow modelling and computation: A survey. *Comput. Vis. Image Underst.* **2015**, *134*, 1–21. [CrossRef]

36. Yuan, G.; Sun, P.; Zhao, J.; Li, D.; Wang, C. A review of moving object trajectory clustering algorithms. *Artif. Intell. Rev.* **2017**, *47*, 123–144. [CrossRef]

37. Lee, L.; Romano, R.; Stein, G. Monitoring activities from multiple video streams: Establishing a common coordinate frame. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 758–767. [CrossRef]

38. Liu, W.; Camps, O.; Sznaier, M. Multi-camera multi-object tracking. *arXiv* **2017**, arXiv:1709.07065.

39. Hatwar, R.B.; Kamble, S.D.; Thakur, N.V.; Kakde, S. A review on moving object detection and tracking methods in video. *Int. J. Pure Appl. Math.* **2018**, *118*, 511–526.

40. Ranković, N.; Radmanović, T.; Ranković, D.; Bjelica, M. Visualization tools for large amount of data. *Appl. Internet Inf. Technol.* **2019**, *122*, 122–127.

41. Schmidt, J. Usage of Visualization Techniques in Data Science Workflows. In Proceedings of the VISIGRAPP (3: IVAPP), Valletta, Malta, 27–29 February 2020; pp. 309–316.

42. De la Cruz Dorantes, C.R.; Thompson, E.F.B.; Vanoye, J.A.R.; Solís, J.A.A. Methodology for the comprehensive application of visualization techniques in data science as support for decision making. *Strateg. Technol. Soc.* **2017**, *5*, 31–50.

43. Cavaller, V. Dimensional taxonomy of data visualization: A proposal from communication sciences tackling complexity. *Front. Res. Metrics Anal.* **2021**, *6*, 643533. [CrossRef] [PubMed]

44. Jena, B. A review on data visualization tools used for big data. *Int. Res. J. Eng. Technol. (IRJET)* **2017**, 4, 492–495.

45. Hajirahimova, M.; Ismayilova, M. Big data visualization: Existing approaches and problems. *Probl. Inf. Technol.* **2018**, *9*, 65–74. [CrossRef]

46. Khatri, A.; Gupta, N. Impact of Data Visualization on Management Decisions. *Lond. J. Res. Manag. Bus.* **2022**, *22*, 53–62.

47. Mourtzis, D.; Siatras, V.; Zogopoulos, V. Augmented reality visualization of production scheduling and monitoring. *Procedia Cirp* **2020**, *88*, 151–156. [CrossRef]

48. Venkatesan, M.; Mohan, H.; Ryan, J.R.; Schürch, C.M.; Nolan, G.P.; Frakes, D.H.; Coskun, A.F. Virtual and augmented reality for biomedical applications. *Cell Rep. Med. 2* **2021**. [CrossRef]

49. Reipschlager, P.; Flemisch, T.; Dachselt, R. Personal augmented reality for information visualization on large interactive displays. *IEEE Trans. Vis. Comput. Graph.* **2020**, *27*, 1182–1192. [CrossRef] [PubMed]

50. Carmigniani, J.; Furht, B.; Anisetti, M.; Ceravolo, P.; Damiani, E.; Ivkovic, M. Augmented reality technologies, systems and applications. *Multimed. Tools Appl.* **2011**, *51*, 341–377. [CrossRef]

51. Christoff, N.; Neshov, N.N.; Tonchev, K.; Manolova, A. Application of a 3D Talking Head as Part of Telecommunication AR, VR, MR System: Systematic Review. *Electronics* **2023**, *12*, 4788. [CrossRef]

52. Azuma, R.T. A survey of augmented reality. *Presence Teleoper. Virtual Environ.* **1997**, *6*, 355–385. [CrossRef]

53. Arafa, A.; Sheerah, H.A.; Alsalamah, S. Emerging Digital Technologies in Healthcare with a Spotlight on Cybersecurity: A Narrative Review. *Information* **2023**, *14*, 640. [CrossRef]

54. Nunes, J.S.; Almeida, F.B.; Silva, L.S.; Santos, V.M.; Santos, A.A.; de Senna, V.; Winkler, I. Three-dimensional coordinate calibration models for augmented reality applications in indoor industrial environments. *Appl. Sci.* **2023**, *13*, 12548. [CrossRef]

55. Bhang, K.J.; Huh, J.R. Effectiveness of Fine Dust Environmental Education on Students' Awareness and Attitudes in Korea and Australia Using AR Technology. *Sustainability* **2023**, *15*, 16039. [CrossRef]

56. Kleftodimos, A.; Evagelou, A.; Gkoutzios, S.; Matsiola, M.; Vrigkas, M.; Yannacopoulou, A.; Triantafillidou, A.; Lappas, G. Creating Location-Based Augmented Reality Games and Immersive Experiences for Touristic Destination Marketing and Education. *Computers* **2023**, *12*, 227. [CrossRef]

57. Stappung, Y.; Aliaga, C.; Cartes, J.; Jego, L.; Reyes-Suárez, J.A.; Barriga, N.A.; Besoain, F. Developing 360° Virtual Tours for Promoting Tourism in Natural Parks in Chile. *Sustainability* **2023**, *15*, 16043. [CrossRef]

58. Martín-Gutiérrez, J.; Mora, C.E.; Añorbe-Díaz, B.; González-Marrero, A. Virtual technologies trends in education. *Eurasia J. Math. Sci. Technol. Educ.* **2017**, *13*, 469–486.

59. Nixon, M.; Aguado, A.S. *Feature Extraction and Image Processing for Computer Vision*; Academic Press: New York, NY, USA, 2012.

60. Sharma, V.; Nain, N.; Badal, T. A survey on moving object detection methods in video surveillance. *Int. Bull. Math. Res.* **2015**, *2*, 209–218.

61. Hernández-Díaz, T.; Vázquez-Cervantes, A.; González-Barbosa, J.J.; Barriga-Rodríguez, L.; Herrera-Navarro, A.M.; Baldenegro-Pérez, L.A.; Jiménez-Hernández, H. Detecting background and foreground with laser array system. *Measurement* **2015**, *63*, 195–206. [CrossRef]

62. Piccardi, M. Background subtraction techniques: A review. In Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583), Hague, The Netherlands, 10–13 October 2004; Volume 4, pp. 3099–3104.

63. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*; Cambridge University Press: Cambridge, UK, 2003.

64. Roushdy, M. Comparative study of edge detection algorithms applying on the grayscale noisy image using morphological filter. *GVIP J.* **2006**, *6*, 17–23.

65. Magdy, N.; Sakr, M.A.; Mostafa, T.; El-Bahnasy, K. Review on trajectory similarity measures. In Proceedings of the 2015 IEEE Seventh International Conference on Intelligent Computing and Information Systems (ICICIS), Cairo, Egypt, 12–14 December 2015; pp. 613–619. [CrossRef]

66. Xie, M. EDS: A segment-based distance measure for sub-trajectory similarity search. In Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, Snowbird, UT, USA, 22–27 June 2014; ACM: New York, NY, USA, 2014; pp. 1609–1610.

67. Alt, H. The computational geometry of comparing shapes. In *Efficient Algorithms*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 235–248.
68. Guo, N.; Ma, M.; Xiong, W.; Chen, L.; Jing, N. An efficient query algorithm for trajectory similarity based on Fréchet distance threshold. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 326. [CrossRef]
69. Faloutsos, C.; Ranganathan, M.; Manolopoulos, Y. *Fast Subsequence Matching in Time-Series Databases*; ACM: New York, NY, USA, 1994; Volume 23, pp. 419–429.
70. Femiani, J.C.; Razdan, A.; Farin, G. Curve Shapes: Comparison and Alignment. Submitted to TPAMI. Available online: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=68c25f7f1b3d7b0dddacad91933a3b4311942ed2 (accessed on 23 November 2023).
71. Huttenlocher, D.P.; Klanderman, G.A.; Rucklidge, W.J. Comparing images using the Hausdorff distance. *IEEE Trans. Pattern Anal. Mach. Intell.* **1993**, *15*, 850–863. [CrossRef]