UNIVERSIDAD NACIONAL DE COLOMBIA
SEDE MEDELLÍN
FACULTAD DE MINAS

DYNA
http://dyna.medellin.unal.edu.co/

Optimized registration based on an ant colony for markerless augmented reality systems

Gloria Elena Jaramillo-Rojas [a] & John Willian Branch-Bedoya [b]

[a] *Universidad de la Costa, Barranquilla, Colombia. gjaramil@cuc.edu.co*
[b] *Facultad de Minas, Universidad Nacional de Colombia, Medellín, Colombia. jwbranch@unal.edu.co*

Received: December 10th, 2019. Received in revised form: February 6th, 2020. Accepted: February 27th, 2020

Abstract
Accurate registration in augmented reality systems is essential to guarantee the visual consistency of the augmented environment. Although error in the virtual-real alignment is almost unavoidable, different approaches have been proposed to quantify and reduce such errors. However, many of the existing solutions require a lot of a priori information, or they only focus on camera calibration to guarantee good results in the registration. This article presents a heuristic method that aims to reduce registration errors in markerless augmented reality systems. The proposed solution sees error reduction as a mono-objective optimization problem, which is addressed by means of the Ant Colony Optimization (ACO) algorithm. Experimental results reveal the validity of the proposed method, reaching an average error of 1.49 pixels for long video sequences.

Keywords: augmented reality; markerless registration; meta-heuristic method; ant colony optimization.

Registro optimizado basado en colonia de hormigas para sistemas de realidad aumentada sin marcadores

Resumen
Un registro preciso en sistemas de realidad aumentada es esencial para garantizar la consistencia visual del entorno aumentado. Aunque el error en la alineación virtual-real es casi inevitable, en la literatura se han propuesto varios enfoques para cuantificar y reducir dicho error. Sin embargo, muchos de los trabajos existentes requieren mucha información a priori, o sólo se centran en la calibración de la cámara para garantizar buenos resultados. En este artículo, se presenta un método meta-heurístico para reducir el error en el registro. Nuestra solución considera la reducción del error como un problema de optimización mono-objetivo, que se aborda mediante el Algoritmo de Colonias de Hormigas (ACO). Los resultados experimentales revelan la validez del método propuesto, alcanzando un error promedio de 1,49 píxeles.

Palabras clave: realidad aumentada; registro sin marcadores; método meta-heurístico; optimización de colonias de hormigas.

1. Introduction

Augmented Reality's (AR) main objective is to improve visual information perceived by the user by adding synthetic objects to the real space. Solving visualization and interaction problems is what motivates the creation of these environments.

In recent decades, a considerable number of applications have been proposed in medicine [1], education [2], museology [3], and entertainment [4] among others. Regardless of the application, AR systems follow a common process (Fig. 1) in order to accurately determine the position of the virtual object in the real space. In the registration step, the virtual information is aligned with the real information to create the augmented environment. A mixed environment such as this should be as coherent as possible so that misalignment be imperceptible during interaction. This is particularly important as AR can be used to support visualization tasks in critical applications such as surgical operations. In these cases, an incorrect registration of even

How to cite: Jaramillo-Rojas, G.E. and Branch-Bedoya, J.W, Optimized Registration based on an Ant Colony for Markerless Augmented Reality Systems. DYNA, 87(212), pp. 259-266, January - March, 2020.

© The author; licensee Universidad Nacional de Colombia.

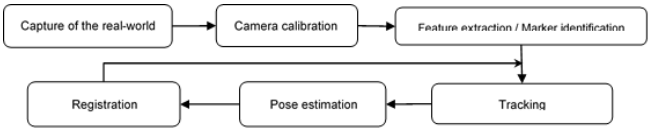

Figure 1. General process of an augmented reality system.
Source: The Authors.

two millimeters would be catastrophic. This situation is exacerbated by the requirement of real-time interaction, which implies the permanent calculation of accurate poses according to changes in the user's point of view, within a reasonable time period.

Due to the great importance of a correct alignment of the virtual objects, different approaches have been proposed to reduce registration error. Nevertheless, many of these solutions require a lot of a priori information, or they only focus on guaranteeing correct camera calibration.

This article presents a method based on a clear methodology to reduce visual inconsistency in markerless augmented reality systems. It aims to calculate the *best* matrix in the affine space by relating 3D points in the world with their 2D projection in the video frame. In practical terms, it becomes a combinatorial problem because only four points— the *n* points extracted from the real scene—are needed to calculate the reprojection matrix. The use of the Ant Colony Optimization meta-heuristic is proposed to select the best points and so reduce the reprojection error. To the best of our knowledge, this is the first time that this meta-heuristic has been implemented to tackle the registration problem.

This article is organized as follows: the literature review is laid out in Section 2. The methodology followed is described in Section 3, and the details of the proposed method in Section 4. Experimental results are discussed in Section 5. Finally, conclusions and future studies are presented in Section 6.

2. Previous studies

The literature review presented below is organized into two sections. The first section reviews the different techniques reported that were used to register virtual-real information. The second section focuses on studies that address the reduction of registration errors.

2.1. Registration techniques

One of the most popular approaches for registration has been the direct placement of artificial marks into a scene. Due to its accuracy, many academic and industrial researchers have focused their attention on this kind of registration. In this technique, virtual objects are directly visualized over the marks. For this reason, it will be easier to determine the position for the superimposition of the virtual object if the mark can easily be recognized in the scene.

In the literature, different designs for marks have been proposed. One of the first works in this field was carried out in 1999 by Kato [5], who introduced the use of binary square marks. This technique is still used in works such as [6-8]. Other approaches highlight the advantages of embedded markers [9] and retroreflectors [10]. In general, using marks in the registration is advantageous in that it simplifies the process of feature extraction and pose calculation. However, such simplification implies the manipulation of the real environment by incorporating fiducials, which Schall [11] has called visual pollution. Additionally, an important limitation to take into account when using marks is occlusion. If portions of the mark are partially hidden, identification might not be carried out successfully, and the registration will fail. To overcome these limitations when using marks, an alternative approach consists of capturing the information for pose calculation from the natural characteristics of the scene.

Markerless registration [12,13] aims to identify features directly from the scene. The challenges that this method generates are associated with the selection of the features to be tracked, robustness when faced with rapid movements, and the calculation of the correct position for the superimposition of the virtual objects. As a result, much more work is needed to obtain the accuracy achieved by artificial marks. Nevertheless, this approach is suitable in environments where the use of marks is not feasible or desirable.

Finally, another set of studies focus on hybrid approaches, which combine visual-based methods with electromagnetic tracking [14,15]. The accuracy of this hybrid registration is achieved through the careful acquisition of the tridimensional coordinates of feature points by using 2D information extracted from the video sequence and sensors. The main disadvantage of this method is that it usually requires a prepared environment.

2.2. Error correction

Much effort and research has gone into the development of a technique accurate enough to achieve permanent visual consistency in the mixed environment, and robust enough to tolerate occlusion and changes in lighting. Calibration techniques model the relationship between the information captured by the camera and the real world. Some studies focus on analyzing an error by paying special attention to careful camera calibration [16], edge detection [17] or feature matching [18]. The use of heuristic methods has also been attempted for registration purposes. In [19], to determine the best pose, generic algorithms are implemented which minimize the 2D points of an image and the reprojection of its tridimensional points. The results show significant error reduction with a low associated computer cost. However, it requires the 3D model of the object used to render the visual information.

Considering the studies discussed above, the method used in this paper aims to overcome the limitations associated with:

- Inaccuracy in the registration for long video sequences.
- The use of a priori information, i.e., artificial markers and

3D models created beforehand.

The innovative element and main contribution of this work is the transformation of the registration problem into a combinatorial optimization problem, which is addressed by using a heuristic method. Below, we describe the methodology that was followed to create the AR system. The implemented methods are based on the Kanade Lucas Tomasi (KLT) algorithm for feature extraction and tracking, together with the calculation of affine reconstruction and reprojection matrices.

3. Methodology

A modular approach was followed for each process depicted in Fig. 1, without losing sight of the interrelation between modules. Each process is described below, although neither the techniques used for the extraction of features nor the optical flow method will be explained in detail since they have already been widely reported in the literature.

3.1. Feature extraction

Due to intrinsic or extrinsic factors, each step shown in Fig. 1 produces a certain degree of error. The error accumulated over the whole process can seriously affect the visual consistency of the AR system. Consequently, the method used for the extraction of natural features cannot be chosen at random. Neumann [20] states that a good characteristic is one which is tracked correctly along a video sequence. Therefore, when selecting relevant features, it is important to take into account that these must possess properties which guarantee stability and reliability.

In this study, the Shi-Tomasi method was chosen as the means of detecting the set of relevant features from the real world. This method uses eigenvalues to detect corners and proposes both a translation and affine model, which defines a point's displacements and appearance changes.

3.2. Tracking

Tracking deserves special attention since it involves two main components, namely, modeling and identification. The former is used for the calculation of the pose and the latter is associated with 2D matching in image sequences. Tracking is considered to be the leading cause of error. This is because much of the information used in this stage comes from estimations, so minor errors can accumulate until the results generated become visually unacceptable in the mixed environment.

For the tracking, a pyramidal sparse optical flow method is implemented which determines the coordinates of where a feature appears in the frame *i+1* regarding the features extracted in frame *i*. The selected tracker tolerates modeling for the relative movement of objects and the observer. Nevertheless, it does not model tridimensional movement of objects; it only models the image changes resulting from such movements.

3.3. Obtaining pose information

To determine position and orientation from features, a rotation and translation matrix is implemented that relates tridimensional points with their bidimensional projections. To calculate 3D points, the Euclidean coordinate system transformed into the affine coordinate system [21] is used. This has been widely employed for AR purposes because of its reprojection and reconstruction properties.

3.3.1. Reconstruction property

Consider an affine space defined by four non-coplanar points $\{P_0, P_1, P_2, P_3\} \in R^3$ where P_0 is the origin of the affine space, and $\{P_1, P_2, P_3\}$ are three basis points. The tridimensional coordinates $[X\ Y\ Z]^T$ of the extracted features can be determined in the affine space if the projections of each feature point and the projections of the four points defining the affine space $\{P_0, P_1, P_2, P_3\}$ are known in two different views ($[u^j\ v^j]^T$; j=1, 2) (see eq. 1).

$$\begin{bmatrix} u^1 \\ v^1 \\ u^2 \\ v^2 \end{bmatrix} = \begin{bmatrix} u^1_{b1} - u^1_{b0} & u^1_{b2} - u^1_{b0} & u^1_{b3} - u^1_{b0} & u^1_{b0} \\ v^1_{b1} - v^1_{b0} & v^1_{b2} - v^1_{b0} & v^1_{b3} - v^1_{b0} & v^1_{b0} \\ u^2_{b1} - u^2_{b0} & u^2_{b2} - u^2_{b0} & u^2_{b3} - u^2_{b0} & u^2_{b0} \\ v^2_{b1} - v^2_{b0} & v^2_{b2} - v^2_{b0} & v^2_{b3} - v^2_{b0} & v^2_{b0} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (1)$$

where projections $[u^j_{bi} - u^j_{bi}]^T$ (j = 1, 2; i = 0, 1, 2, 3) are the affine origin points and the basis points in the two images.

In order to compute Eq. 1, two reference images (*RI*1 and *RI*2) need to be extracted from the video sequence. To reduce computational time, the feature extraction process is carried out only once in *RI*1 and then tracked with KLT to *RI*2. This process continues during the entire video sequence, where the old *RI(k + 1)* image becomes the new *RI(k)*, and the new captured frame from the sequence video becomes the new *RI(K + 1)*.

3.3.2. Reprojection property

This property calculates the projection [*u v*] of a point in an image *k* by taking into account its affine 3D coordinate, the base projection and the basis points (see eq. 2).

$$\begin{bmatrix} u_k \\ v_k \\ 1 \end{bmatrix} = \begin{bmatrix} u_{b1} - u_{b0} & u_{b2} - u_{b0} & u_{b3} - u_{b0} & u_{b0} \\ v_{b1} - v_{b0} & v_{b2} - v_{b0} & v_{b3} - v_{b0} & v_{b0} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_k \\ Y_k \\ Z_k \\ 1 \end{bmatrix} \quad (2)$$

Due to the fact that in this proposition, the 2D coordinates of relevant features are obtained using the KLT tracker, the reprojection matrix will be used to compute the error by comparing the information obtained by the tracker with the information given by the matrix.

In this study, it is assumed that virtual objects will be superimposed on planar surfaces. So, when the system initializes, the user is asked to manually select four points to define the space used for the rendering of the virtual object. Note that in doing this, the four points used to define the

affine space cannot be used because of the non-coplanarity restriction. The selection of these additional points is not a disadvantage of the method. On the contrary, it means that there is a greater tolerance of occlusions, since, if a point is hidden, it will be possible to estimate its coordinates from the reprojection property as explained in [22].

On the other hand, the definition of the basis points and the origin of the affine coordinate system is automatically carried out by using the method proposed in [22] consisting of a Singular Value Decomposition (SVD) of a matrix A which contains the coordinates of the feature points of the image and is situated in the center of mass of the feature points (see eq. 3).

$$A_{4xn} = \begin{bmatrix} u_{p1}^1 - u_c^1 & \cdots & u_{pn}^1 - u_c^1 \\ v_{p1}^1 - v_c^1 & \vdots\vdots & v_{pn}^1 - v_c^1 \\ u_{p1}^2 - u_c^2 & \cdots & u_{pn}^2 - u_c^2 \\ v_{p1}^2 - v_c^2 & & v_{pn}^2 - v_c^2 \end{bmatrix} \quad (3)$$

Where $[u_{pi}^j \quad v_{pi}^j]$ (j=1, 2; i=1, 2,. . . , n) represents the projection of the 3D affine points in the two different views and $[u_c^j \quad v_c^j]$ (j=1, 2) is the center of mass of these projections. By SVD decomposition, the matrix A is expressed in eq. 4 in the form:

$$A_{4xn} = U_{4x4} D_{4xn} V_{nxn}^T \quad (4)$$

where U_{4x3}, D_{3x3} and V_{nx3} are the upper sub-matrices 4x3, 3x3 and nx3 of U, D, and V respectively. With this factorization, reconstruction and reprojection matrices can be formulated by eq. 5 and 6 respectively.

$$U_{4x3} D_{3x3}^{1/2} = \begin{bmatrix} u_{b1}^1 - u_c^1 & u_{b2}^1 - u_c^1 & u_{b3}^1 - u_c^1 \\ v_{b1}^1 - v_c^1 & v_{b2}^1 - v_c^1 & v_{b3}^1 - v_c^1 \\ u_{b1}^2 - u_c^2 & u_{b2}^2 - u_c^2 & u_{b3}^2 - u_c^2 \\ v_{b1}^2 - v_c^2 & v_{b2}^2 - v_c^2 & v_{b3}^2 - v_c^2 \end{bmatrix} \quad (5)$$

$$M_{3x4}^k = \begin{bmatrix} u_{b1}^k - u_c^k & u_{b2}^k - u_c^k & u_{b3}^k - u_c^k & u_c^k \\ v_{b1}^k - v_c^k & v_{b2}^k - v_c^k & v_{b3}^k - v_c^k & v_c^k \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

Finally, affine 3D points are obtained with eq. 7.

$$[P_1 \quad \ldots \quad P_n] = \begin{bmatrix} D_{3x3}^{1/2} V_{nx3}^T \\ 1 \end{bmatrix} \quad (7)$$

3.3.3. Error calculation

Considering that the reprojection matrix relates the n feature points with their tridimensional points (see eq. 8), and that matrix M_{3x4} can be used to obtain 2D projections of any point when its tridimensional coordinates are known, it is possible to calculate the bidimensional coordinates of the four points selected by the user to visualize the virtual object (see Section 3.3.2), if their tridimensional coordinates are known. This is carried out by calculating the reconstruction matrix from the factorization of the matrix A.

$$2DFeatures_{3xn} = M_{3x4} 3DFeatures_{4xn} \quad (8)$$

The reprojection matrix is used to calculate the error using eq. 9; where $\widehat{m_{kl}}$ are the coordinates estimated with the matrix, and m_{ki} are the coordinates given by the tracker.

$$error = \frac{\sum_{i=1}^{n} \| m_{ki} - \widehat{m_{kl}} \|}{n} \quad (9)$$

This study aims to find the best M_{3x4} matrix to minimize the error between the two data sets. In the described affine equation system, a problem becomes combinatorial when only four points are required to solve eq. 8. So, it will be necessary to find the best combination of 2D/3D point pairs among the set of extracted features which minimize the reprojection error. Consequently, eq. 9 becomes the target to be minimized, i.e., the mono-objective function for the optimization algorithm (see Section 4).

The justification for the implementation of a heuristic method is twofold:

- A combinatorial optimization method enables the selection of only the best four points from the total set of extracted features.
- Combinatorial problems are NP complete, which computing time is unacceptable in AR applications where real time interaction is required. Therefore, a meta-heuristic method enables an estimation of the optimal solution within an acceptable computing time.

4. The error reduction as an optimization problem

In 1959, the entomologist Pierre-Paul Grasse observed that termites reacted to certain stimulus. This had the effect of acting as new stimulus for the insect which produced it as well as for the colony as a whole. Grasse used the term stigmergy to describe this communication among animals of the same species. Stigmergy examples are evident in ant colonies where they move from and to a food source by depositing pheromones along the way. Other ants in the colony tend to follow the path where they perceive a strong pheromone track.

In Ant Colony Optimization (ACO), artificial ants move within a graph which contains the search space of the problem. Each ant is considered to be a possible solution to the problem. In the graph, the ant walks from node to node, building a solution, with the restriction that a node cannot be visited twice by the same ant. For each iteration, the ant chooses its which node to visit next depending on the amount of pheromone of the nodes available to be visited, where a higher degree of pheromone represents a higher probability of visiting that node. After finishing the iteration, pheromone values are modified according to the quality of the solutions built by the ants in each iteration.

Optimization based on ants was formalized as heuristic by Dorigo et al. [23], who described the following components for a model $P = \{S, \Omega, f\}$ of a combinatorial

optimization problem.

- A search space S defined over a finite group of discrete decision variables X_i
- A restriction group Ω over the variables.
- An objective function $f = S \rightarrow R_0^+$ to be minimized.

The generic variable X_i takes values in $D_i = \{v_i^1, \ldots, v_i^{(|D_i|)}\}$. A possible solution s ϵ S is a complete assignment of variable values which satisfies all restrictions Ω. A solution s* ϵ S is called the global optimum if and only if $f(s^*) \leq f(s) \forall s \in S$.

Using the ACO model, this study aims to calculate the reprojection matrix that will minimize the error between the estimated and the real coordinates. It will obtain, as accurately as possible, the two-dimensional positions of points where the virtual objects will be superimposed. To do this, the error is established as the objective function according to eq. 9. The decision variables are represented by the set of features tracked by KLT. In the present case, the model is expressed as follows:

- $S = \{X_1, X_2, \ldots, X_i\}$; where $i = 1$ to the number of features, and $X = (u, v)$
- f = error = $\frac{\sum_{i=1}^{n} \|m_{ki} - \widehat{m_{ki}}\|}{n}$

Although, in the literature, several methods based on ant optimization have been proposed, we choose the Ant System (AS) method which is based on the permanent updating of the pheromones for all m ants that have built a solution. The total pheromone of each node is computed according to eq. 10.

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \tag{10}$$

where (1 - ρ) is an evaporation rate and $\Delta\tau_{ij}^{k}$ is the pheromone track which is computed as indicated in eq. 11:

$$\Delta\tau_{ij}^{k} = \begin{cases} \frac{Q}{L_k} & if\ feature \in S^k \\ 0 & otherwise \end{cases} \tag{11}$$

where Q is a constant and L_k is the function to be optimized. The selection of the next node to be visited by an ant is a stochastic process given by eq. 12.

$$p_{ij}^{k} = \begin{cases} \frac{\tau_{ij}^{\alpha}\eta_{il}^{\beta}}{\sum_{c_{il} \in N(s^P)} \tau_{il}^{\alpha}\eta_{il}^{\beta}} & if\ c_{il} \in N(s^P) \\ 0 & otherwise \end{cases} \tag{12}$$

where s^P is a partial solution, $N(s^P)$ are the possible nodes to be selected and l is a node not yet visited by the ant k. Finally, η_{il}^{β} is obtained with eq. 13.

$$\eta_{il}^{\beta} = \frac{1}{d_{ij}} \tag{13}$$

d_{ij} is a function that describes the performance of the solution when the node j has been incorporated. For each iteration, the feature which is selected next will be the one which obtains the maximum value when eq. 14 is applied.

$$\text{NextFeature}^{s_j} = \begin{cases} \frac{\text{TotalPheromone}_j^{\alpha}}{\sum_{g \notin s_j} \text{TotalPheromone}_g^{\alpha}} \end{cases} \tag{14}$$

Algorithm 1 presents the methodology which clarifies how ACO should be used for the selection of the best features. The points (represented by an ant) are used to calculate the reprojection matrix which minimizes the registration error. This means that the matrix—according to ACO—accurately reprojects the extracting points. Algorithm 2 presents the complete algorithm for the whole process.

Algorithm 1. ACO for the selection of the best features

```
foreach ant do
    Its pheromone value is set to zero
    Assign four features randomly
    Calculate the error according to Eq. 9
end
Sort ants according the error measure
bestAnts ← Select the numBestAnts best ants
for k ← 1 to numBestAnts do
    foreach feature f in ant do
        pheroTrack_f ← Calculate the pheronome track
        totalPheromone_f ← cte * totalPheromone_f + pheroTrack_f
    end
end
while i < numIterations do
    for u ← 1 to numAnts do
        //A partial solution
        S_u ← Generate the bestPartial from bestAnts
    end
    for r ← bestPartial + 1 to 4 do
        for h ←1 to numAnts do
            Select the following feature according to Eq. 14
        end
    end
    for j ←1 to numAnts do
        Calculate the error according to Eq. 9
    end
    Sort ants according the error measure
    bestAnts ← Select the numBestAnts best ants
    for k ← 1 to numBestAnts do
        foreach feature f in ant do
            pheroTrack_f ← Calculate the pheromone track
            totalPheromone_f ← cte * totalPheromone_f + pheroTrack_f
        end
    end
end
return bestAnts
```

5. Results and discussion

In general, with ACO, an ant builds a solution by selecting 4 points from which the M_{3x4} matrix is calculated. This matrix is used to calculate the two-dimensional projections and then compare them with those made by the KLT tracker, by using the objective function described in eq. 9. When an ant presents a good solution (a low error), the four points it contains will have a greater probability (greater pheromone) of being chosen in the next iteration.

Tests were made with nine different video sequences, which were between 1710 and 3510 frames long. Results show an average of 0.5% of features lost along the nine videos. Note that the percentage of features which are not correctly tracked are of great importance. This is because the 3D coordinates of the features are obtained from the reconstruction matrix which was obtained from factoring matrix A. If some of these points are lost, matrix A, its center of mass, and the matrices *U, V, D*, will change. Therefore, projections and reconstructions will not accurately represent the structure of the scene. In the system proposed by this study, the Euclidean space is built from the four points manually chosen by the user. This task is carried out after the features are extracted. At the same time, the reference images (*RI*1 and *RI*2) are captured. They are used to define the affine space. Results of this stage are shown in Fig. 2.

Algorithm 2. Complete Algorithm

```
Capture frame1
trackedFeatures = fnExtractFeatures(frame1) //Shi-Tomasi
while true do
    Capture frame2
    //KLT
    trackedFeatures ← fnTrack(frame1, frame, trackedFeatures)
    //user choses a frame to define the Euclidean space
    if event then
        //3D coord, reconstruction and reprojection matrices
        ACS = fcDefineACS(trackedFeatures)
        // define the Euclidean space in RI1 and RI2
        defineWCS(frame1, frame2)
        //3D Euclidean space
        TridiWCS = fnCreateTridWCS(ACS)
        createdWCS = true
    end
    if createdWCS then
        //calculates the best reprojection matrix
        ReproMatrix = ACO(trackedFeatures, ACS)
        //Optimized 2D points of the Euclidean space
        Repro2DWCS = fcReproWCS(ReproMatrix, TridiWCS)
        drawReprojection(Repro2DWCS, frame1, frame2)
    end
end
```

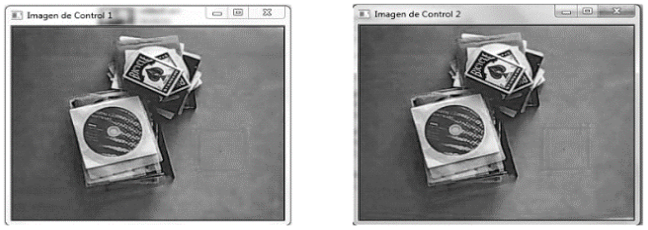

Figure 1. Definition of the Euclidean space (left) Image IR1 (right) Image IR2
Source: The Authors.

To select the points that form the affine coordinate system, the user can use any mark that allows the four points in both images to be identified correctly. This pattern should be removed from the scene after the points have been chosen.

After the affine space is calculated, the reconstruction and reprojection matrices, the 3D coordinates of features, and the number of tracked features are computed. If the number of tracked features change, the process must be restarted. Otherwise, the 3D coordinates of the points forming the Euclidean space are obtained by relating the reconstruction matrix with the bidimensional coordinates chosen by the user.

At time $t = k$, the reprojection matrix is calculated by solving the system of equations relating the 3D points of features to their corresponding 2D projections given by the tracker. Considering that the ACO method was used to determine the best reprojection matrix, it can be used to reproject the points of the Euclidean space with the lowest error.

Results in the video sequences are shown in Figs. 3 and 4. The region used for registration is presented in red. Note that in the latter, the mark was removed after choosing the points which form the Euclidean space.

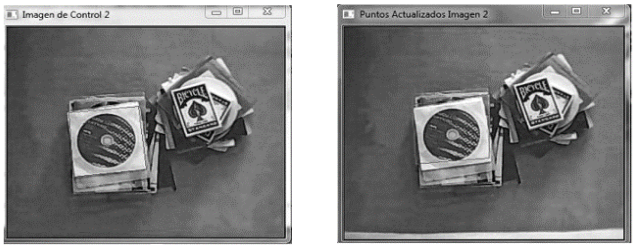

Figure 2. Results with ACO (left) RI1 (right) frame 132
Source: The Authors.

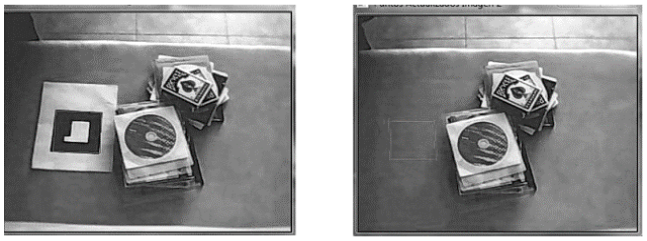

Figure 3. Results with ACO (left) frame 298 (right) frame 405
Source: The Authors

Table 1.
Parameters of the proposed method.

Parameter	Value
Number of features	15
Number of ants	200
Number of iterations	12
Evaporation rate	0.25
α	1

Source: The Authors.

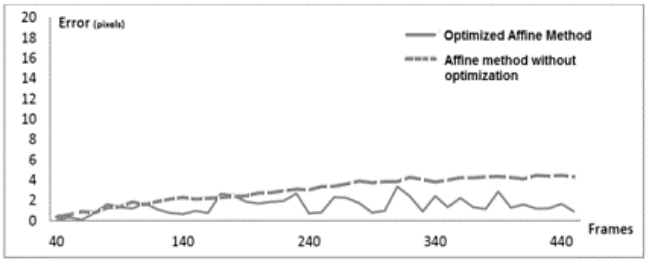

Figure 4. Comparative Results
Source: The Authors.

Tests were carried out where 15 features were considered within video sequences of over 350 frames. In the implementation of the optimized affine method, 200 ants were created, and 12 iterations were carried out. The error measures are plotted in Fig. 5 for 440 frames.

The comparison of the method proposed here with Pang's proposition revels that the latter is more stable than the proposed one. However, the registration based on Ant Colony does not suffer from accumulated errors since it does not use previous positioning results. Moreover, the selection of the number of iteration and features is not a trivial step. Augmented reality systems run in real-time, so the optimization algorithm must converge before the processing of the next pair of frames. It is important to highlight that not every single frame was processed since the differences between two consecutive frames are not significant for the tracking process. Therefore, taking into account that ACO always finds a minimum, even if it is not the global minimum, the number of iterations was chosen as a stopping criterion, and was set to 12. For the present proposition, the number of features necessary for the convergence of the algorithm within the required time, and the number of iterations were both set empirically.

6. Conclusions and future studies

In this article, a methodology based on the Ant Colony Optimization (ACO) method is presented. ACO is proposed as a method to select the best set of points used to calculate the reprojection matrix that will generate the minimum error margin in the calculation of the 2D coordinates in which the virtual object is rendered. The selection of the best set of points is clearly a combinatorial optimization problem, which needs to be solved within a reasonable time frame because of the real-time interaction required in AR systems.

This proposal contributes to the improvement of visual alignment in markerless AR environments. The general performance of the method showed an average error of 1.49 pixels, with a standard deviation of 0.748. Moreover, an average reduction of 49.52% in the positioning error was obtained, compared to a non-optimized affine method. Although this method yields good results, new challenges emerge. Firstly, a feedback technique needs to be implemented to reduce the instability in the proposed method (jitters in Fig. 5). Second, the restriction of rendering on planar surfaces needs to be relaxed. Currently, we are working on the comparison of the proposed approach with other methods, notably, black hole. Similarly, we are working on the automatic setting of parameters.

References

[1] Sutherland, J., Belec, J., Sheikh, A, Chepelev, L., Althobaity, W., Chow, B.J.W., Mitsouras, D., Christensen, A., Rybicki, F.J. and La Russa, D.J., Applying modern virtual and augmented reality technologies to medical images and models. Journal of Digital Imaging, 32(1), pp. 38-53, 2019. DOI: 10.1007/s10278-018-0122-7

[2] Dalim, C.S.C., Piumsomboon, T., Dey, T. Billinghurst, M. and Sunar, S., TeachAR: An interactive augmented reality tool for teaching basic english to non-native children. In: IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct), 2016, pp. 344-345. DOI: 10.1109/ISMAR-Adjunct.2016.0046

[3] Jaramillo, G.E., Quiroz, J.E., Cartagena, C.A., Vivares, C.A. and Branch, J.W., Mobile Augmented reality applications in daily environments. Revista EIA, - Escuela de Ingeniería de Antioquia. 14, pp. 125-134, 2010. ISSN 1794-1237.

[4] Kadi, D., Suyoto and Santoso, A.J., Mobile application development with augmented reality for promoting tourism objects in Southwest Sumba. In: 3rd International Conference on Science in Information Technology (ICSITech), 2017, pp. 200-205. DOI: 10.1109/ICSITech.2017.8257110

[5] Kato, H. and Billinghurst, M., Marker tracking and HMD calibration for a video-based augmented reality conferencing system. In: Proceedings 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99), 1999, pp. 85-94. DOI: 10.1109/IWAR.1999.803809

[6] Dave, I.R, Chaudhary, V. and Upla, K.P., Simulation of analytical chemistry experiments on augmented reality platform. In: Panigrahi, C., Pujari, A., Misra, S., Pati, B., Li, K.C., eds., Progress in advanced computing and intelligent engineering. Advances in Intelligent Systems and Computing, vol 714. Springer, Singapore, 2019, pp. 393-403. DOI: 10.1007/978-981-13-0224-4_35

[7] Dash, A.K., Behera, S.H., Dogra, D.P. and Roy, P.P., Designing of marker-based augmented reality learning environment for kids using convolutional neural network architecture. Displays, 55, pp. 46-54, 2018. DOI: 10.1016/j.displa.2018.10.003

[8] Yu, X., Yang, G., Jones, S. and Saniie, J., AR marker aided obstacle localization system for assisting visually impaired. IEEE International Conference on Electro Information Technology, pp. 271-276, 2018. DOI: 10.1109/EIT.2018.8500166

[9] Asayama, H., Iwai, D. and Sato, K., Fabricating diminishable visual markers for geometric registration in projection mapping. IEEE Transactions on Visualization and Computer Graphics, 24(2), pp. 1091-1102, 2018. DOI: 10.1109/TVCG.2017.2657634

[10] Oku, H., Uji, T., Zhang, Y. and Shibahara, K., Edible fiducial marker made of edible retroreflector. Computers & Graphics, 77, pp. 156-165, 2018. DOI: 10.1016/j.cag.2018.10.002

[11] Schall, G., Newman, J. and Schmalstieg, D., Rapid and accurate deployment of fiducial markers for augmented reality. In: 10th Computer Vistion Winter Workshop, 2005.

[12] Egodagamage, R. and Tuceryan, M., Distributed monocular visual SLAM as a basis for a collaborative augmented reality framework.

Computers & Graph., 71, pp. 113-123, 2018. DOI: 10.1016/j.cag.2018.01.002
[13] Vlaminck, M., Luong, H. and Philips, W., A markerless 3D tracking approach for augmented reality applications. In: International Conference on 3D Immersion (IC3D), Brussels, 2017, pp. 1-7. DOI: 10.1109/IC3D.2017.8251909
[14] Basori, A.H., Afif, F.N., Almazyad, A.S., Jabal, H.A.S., Rehman, A. and Alkawaz, M.H., Fast markerless tracking for augmented reality in planar environment. 3D Research, 6(41), art. 72, 2015. DOI: 10.1007/s13319-015-0072-5
[15] Luó, X., Feuerstein, M., Kitasaka, T. and Mori, K., Robust bronchoscope motion tracking using sequential Monte Carlo methods in navigated bronchoscopy: dynamic phantom and patient validation. International Journal of Computer Assisted Radiology and Surgery. 7, pp. 371-387, 2012. DOI: 10.1007/s11548-011-0645-6
[16] Moser, K.R., Quantification of error from system and environmental sources in Optical See-Through head mounted display calibration methods, in: 2014 IEEE Virtual Reality (VR), Minneapolis, MN, 2014, pp. 137-138. DOI: 10.1109/VR.2014.6802089
[17] Belhaoua, A., Kornmann, A. and Radoux, J-P., Accuracy analysis of an augmented reality system. In: 12th International Conference on Signal Processing (ICSP), 2014, pp. 1169-1174. DOI: 10.1109/ICOSP.2014.7015184
[18] Du, Y., Miao, Z. and Cen, Y., Markerless augmented reality registration algorithm based on ORB. In: 12th International Conference on Signal Processing (ICSP), 2014, pp. 1236-1240. DOI: 10.1109/ICOSP.2014.7015197
[19] Yu, Y.K., Wong, K.H. and Chang, M.M.Y., Pose estimation for augmented reality applications using genetic algorithm. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics). 35(6), pp. 1295-1301, 2005. DOI: 10.1109/TSMCB.2005.850164
[20] Neumann U. and You, S., Natural feature tracking for augmented reality. IEEE Transactions on Multimedia, 1(1), pp. 53-64, 1999. DOI: 10.1109/6046.748171
[21] Chen, P. and Guan, T., Affine reprojection based registration method for augmented reality. Journal of Computer-Aided Design and Computer Graphics, 22(3), pp. 480-486, 2010.
[22] Pang, Y., Yuan, M.L., Nee, A.Y.C., Ong, S.K. and Youcef-Toumi, K., A Markerless registration method for augmented reality based on affine properties. In: Proceedings of the 7th Australasian User Interface Conference, 50, 2006, pp. 15-22.
[23] Dorigo, M. and Di Caro, G., Ant colony optimization: a new meta-heuristic. In: Proceedings of the Congress on Evolutionary Computation-CEC99, 1999, pp. 1470-1477. DOI: 10.1109/CEC.1999.782657

G.E. Jaramillo-Rojas, received a MSc. in Engineering in Computer Science from the Universidad Nacional de Colombia and a PhD. in Informatics in 2016, from the Université de Pau et des Pays de l'Adour (UPPA), France. Between 2017 and 2018, she worked as a Postdoctoral fellow at Le Conservatoire National des Arts et Métiers de Paris (CNAM), France. Her research interests are focused on computational intelligence techniques.
ORCID: 0000-0002-8830-7247

J.W. Branch-Bedoya, received a BSc. Eng in Mining and Metallurgy Engineering in 1995, an MSc. in Systems Engineering in 1997, and a PhD in Systems Engineering in 2007, all of them from Universidad Nacional de Colombia in Medellin, Colombia. Currently he is a full professor at the Computing and Decision Sciences Department, in the Facultad de Minas, Universidad Nacional de Colombia, Medellin, Colombia. His research interests include: automation, computer vision, digital image processing and computational intelligence techniques.
ORCID: 0000-0002-0378-028X

UNIVERSIDAD NACIONAL DE COLOMBIA
SEDE MEDELLÍN
FACULTAD DE MINAS

Área Curricular de Ingeniería
de Sistemas e Informática

Oferta de Posgrados

Doctorado en Ingeniería- Sistemas e Informática
Maestría en Ingeniería - Analítica
Maestría en Ingeniería - Ingeniería de Sistemas
Maestría en Ingeniería – Sistemas Energéticos
Especialización en Sistemas
Especialización en Mercados de Energía
Especialización en Ingeniería de software
Especialización en Analítica
Especialización en Inteligencia Artificial

Mayor información:

E-mail: acsei_med@unal.edu.co
Teléfono: (57-4) 425 5365