Abstract

One of the major problems in Augmented Reality (AR) is tracking and registration of both cameras and objects. These tasks must be done accurately to combine real and rendered scenes. In particular, the initialization of the object tracking often remains manual in most systems. This paper proposes the use of Bayesian Networks to perform the object recognition and initialization of the tracking. By recognizing the object, special points are taken and we use this information to create generic markers around the scene. Then, an algorithm for pose estimation based on the AR-Toolkit Library is used to find the orientation of the real object to allow the registration process for 3D objects.

1 Introduction

Tracking and registration of both cameras and objects in AR systems are required because, to combine real and rendered scenes, we must project graphical data representations at the right location in real scenes [3]. Recently, visual tracking of image features was used to establish camera pose and register the virtual object with the camera view [17, 11].

In this paper, our emphasis is on the object recognition through natural features such as color and shape for Augmented Reality applications. These objects of interest are previously recorded on an object database. Our recognition process is performed using a Bayesian Network which combines the features related above. Once the object geometry is recovered from the model database, we perform feature detection and extraction from a video sequence. The correspondence between the model features and image features is established based on the same network using some recovered points.

This paper is organized as follows. An overview of related works and object recognition is provided in Section 2. Section 3 discusses our Bayesian network approach followed by our proposal in Section 4. Results are given in Section 5 and final comments in the Section 6.

2 Related Works

Many different sensors can be used to achieve registration in AR systems such as mechanic or magnetic ones, GPS, compass and so on. However these sensors brings some problems like extensive calibration, restricted user displacements, perturbations from the environment and poor accuracy. Some of the previous problems can be solved by using vision based systems. Vision based registration does not require any instrument except the acquiring camera and the augmentation results are generally more accurate than the results obtained by sensors.

Some vision based systems use fiducial markers in the environment to perform the registration task and require the manual calibration of the camera like in [9, 2]. There are also a few efforts addressing markerless tracking in the literature [6, 21, 14]. Due to several reasons (initialization of the system without fiducial markers, image processing in real-time, etc), markerless tracking with mobile cameras is still one of the currently most challenging tasks in augmented reality, and one of the main crucial problem is the automated initialization.

The system proposed in [12] is an automatic initialization method that relies on a learning stage, where a database of key features is constructed based on a set of key frames taken during an offline procedure. The key features consist of a 3D point on the object model and a viewpoint invariant local descriptor based on its appearance in the images. The initialization is done by robustly matching feature points in the initial image with the points present in the database based on a similarity measure. A disadvantage of this method is that these local descriptors are sensitive to scale and zooming. Therefore the working space is limited in the tracking area that is covered by sufficiently enough
key frames.

In the system proposed in [6], a general learning-based framework for feature-based tracking using a single camera was proposed. In a two stage process first a set of natural 3D features is learned using an external tracking system (e.g. marker-based). In the second stage the system uses these learned features for tracking as soon as enough stable features are acquired in the first stage. This marker-less tracking system needs an initialization that provides a rough estimate of the cameras position and orientation. It makes use of an external marker-based tracker for initialization. Once the system loses track it needs to be re-initialized in the same way. In this case the initialization process does not need to be very accurate and perform in real-time. The system is able to converge even with partial or imprecise tracking information for initializing system. However, the authors confirm that using markers for initialization is not an acceptable solution in most applications.

In [13], an automated initialization approach for indoor as well as outdoor environments was proposed. In this system, the initial positional data can be provided by stationary cameras in closed buildings and for instance by GPS for outdoors while providing none or only rudimentary estimates with respect to the user’s orientation. The complete initialization is then achieved by fusing the data from multiple sensors, e.g. mobile and stationary cameras or GPS. In cases where tracking is lost for instance because of large occlusions, the initialization procedure is started automatically. This approach can be easily extended to arbitrary additional sensors.

In this work we propose and presents some results of the use of the Bayesian Network Model to perform the initialization and the tracking phases of an Augmented Reality System. This concept was firstly proposed in [20]. The main contribution of this work is the presentation of some results and implementation issues of this concept. The main advantage of this model is the ability of recording object features and supporting object recognition in the same architecture.

The basic principle underlying the proposed Bayesian Network model follows the general epistemological line presented in literature. Ribeiro-Neto [16] and Silva [19] stated a model for text information retrieval which uses as information content terms such as metadata, www links and text passages (paragraphs or sequences of paragraphs related to the information content), and others. This is a three-layer model: namely, a query layer, a terms layer and an image layer. The query layer carries query terms, the terms layer carries the terms which are possible to occur and the image layer carries the images of the database. In our work, the model proposed by Ribeiro-Neto [16] and Silva [19] is adapted and extended for the tracking problem.

Some works in the literature addressed the use of Bayesian Networks to perform object recognition. In [10, 4], a general framework to build a task oriented 3D object recognition system for CAD based vision using Bayesian Network was proposed. Buxton [7] used object recognition for a visual surveillance system and [22] used Bayesian Networks to classifying a color image of an outdoor scene.

### 3 The Bayesian Network Model

Bayesian Networks are graphical tools for representing causal dependencies between random variables of a joint probability distribution. They are modeled as an acyclic oriented graph, where the random variables are represented by the nodes, also called events, and the relationships between them by the arcs. These relationships stand for causal dependencies between the variables. The strengths of these relationships are shown in tables called Conditional Probability Tables (CPT). A CPT lists the probability that the child node takes on each of its different values for each combination of values of its parents.

The fundamental principle underlying the Bayesian Networks can be stated as follows: since known dependencies between random variables are explicitly declared in the network, the joint probability distribution may be inferred from these dependencies.

Let $X$ and $Y$ be the variables represented by nodes of a Bayesian Network, and let $x$ and $y$ be their respective values. If the $X$ value is a direct cause in the $Y$ value, an arc, oriented from $X$ to $Y$, is added to the graph ($X$ is said “father of $Y$”). The strength of the relationship between $X$ and $Y$ is represented by the conditional probability $P(y|x)$. It is said probability of $Y = y$, in case of $X = x$.

Let $BN$ be a Bayesian Network, Let $Y$ be a random variable in $BN$, let $y$ be the $Y$ value, let $P_X$ be the set of father variables of $Y$ and let $p_y$ be the set of $P_Y$ values. The influence of $P_Y$ on $Y$ may be specified by a function $F$ that satisfies:

$$\sum_{y \in Y} F(y, p_y) = 1 \quad 0 \leq F(y, p_y) \leq 1. \quad (1)$$

The function $F(y, p_y)$ provides a quantification of the conditional probability $P(y|p_y)$. This specification represents the joint probability distribution of the nodes of the $BN$ [15].

In order to exemplify, consider Fig. 1, which represents the joint probability distribution $P(x_1, x_2, x_3, x_4, x_5)$ for the variables $\{X_1, X_2, X_3, X_4, X_5\}$, where $\{x_1, x_2, x_3, x_4, x_5\}$ are their respective values. The $X_1$ node is called network root (node without father) and is the father of $X_2$ and $X_3$. The $P(x_1)$ probability, associated to the $x_1$ value of the root node $X_1$, is called a priori probability and is used to represent the previous knowledge about the problem domain. Given the value of the variable
X1, the X2 and X3 variables are independent. Given the values of the variables X2 and X3, the variables X4 and X5 are independent. Due to these independencies, the joint probability distribution \( P(x_1, x_2, x_3, x_4, x_5) \) may be calculated by:

\[
P(x_1, x_2, x_3, x_4, x_5) = P(x_1)P(x_2|x_1)P(x_3|x_1)P(x_4|x_2, x_3)P(x_5|x_3) \tag{2}
\]

Note that if \( X_1 \) has no parents, then the CPT reduces to unconditional probabilities \( P(X_1) \). For the Directed Acyclic Graph in Fig. 1, the prior probability \( P(X_1) \) must be specified. It has been claimed that prior probabilities are an unwanted introduction of bias to the model, and calculation have been invented in order to avoid it. However, as highlights Jensen [8], prior probabilities are necessary not for mathematical reasons but because prior certainty assessments are an integral part of human reasoning about certainty.

The main advantage underlying the use of Bayesian Networks arises from the fact that it is possible to get implicit information from explicit ones, declared in the graph. For example, consider the nodes in Fig. 1: it is possible to compute \( P(X_5|X_1) \) even this information is not explicitly at a CPT. From Equation (3) and taking advantage of distribution and associative probabilities rules (for details see [8]) we can achieve probabilities such as: \( P(X_3), P(X_4|X_1), P(X_5|X_2), \) etc.

The drawback is that the algorithms have normally a exponential complexity proportional to nodes into the network. However, some algorithms which have theoretical exponential behaviour, in the worst case, also have polynomial performance in most of practical situations.

The model stressed in this paper follows the Bayesian approach, since each CPT is based on a similarity measure between two images, which depend on their content features (we discuss this in detail later). This similarity stands for the degree of believing, according to some criterion, of how similar are these two images. A comment: this is the reason of why some authors call this kind of network a “Believe Network”.

### 4 The Proposed Model

In this work, we use the color and shape features for matching the model object and the target model regions. In the case of color features, consider \( P(O_i|O_m) \) is the probability of the Target Object \( (O_i) \) to occur in the frame \( i \) since the Model Object \( (O_m) \) has occurred in the frame \( i-1 \). Let \( \phi = \{r_1 + \delta_1, g_i + \delta_2, b_i + \delta_3\} \) a range of RGB values and \( \delta_1, \delta_2, \delta_3 \) their variation. In this paper \( P(O_i|O_m) = 1.0 \) if \( \forall p_m(i,j) \in O_m \), also \( p_m(i,j) \in \phi \) and \( \forall p_t(i,j) \in O_t \) also \( p_t(i,j) \in \phi \), where \( p_m \) and \( p_t \) are pixel values. Otherwise, \( P(O_i|O_m) = 0 \). The use of Bayesian Networks composed by color and shape features for Augmented Reality systems is the main contribution of this work.

Our proposed approach has only two layers: the first corresponds to father-nodes (priori information); and the second corresponds to children-nodes (posteriori information). Father-nodes, denoted by \( k_i \).

An oriented arc from an element \( k_i \in \phi \) (term) to an element \( O_j \) (object) stands for the probability of occurrence of the object \( O_j \), once the term \( k_i \) has occurred in the model object. This model can be seen in Fig. 2. The same idea can be applied under the shape features.

The conditional probability that object \( O_j \) occurs given that Model Object \( O_m \) has occurred, can be denoted as:

\[
P(O_j|O_m) = P(O_j|k_1, k_2, ..., k_n) \tag{3}
\]

In this paper, we present a suggestion for the calculation of Equation (3), which is a similarity measure (according to the Bayesian approach) for direct matching between two regions; namely, a Target Object and a Model Object. In order to accomplish this task, we think about the Equation (3) as being an integration of the involved evidences (color
and shape). On the other hand, if we consider the occurrence of a Target Object as being an event conditioned to the occurrence of a Model Object terms, \( k_i \), we may state the Equation (3) as follows:

\[
P(O_j|O_m) = P(O_j|k_1^e, k_2^e, ..., k_n^e, k_1^f, k_2^f, ..., k_m^f)
\]  

(4)

where the variable \( k_i^e \) stands for a color space term, and \( k_i^f \) for a shape space one. For the sake of notation, we make \( KC = \{k_1^e, k_2^e, ..., k_n^e\} \) and \( KF = \{k_1^f, k_2^f, ..., k_m^f\} \), and rewrite the Equation (4):

\[
P(O_j|O_m) = P(O_j|KC, KF)
\]  

(5)

As we have already pointed out, Ribeiro-Neto [16], Silva [19] and Coelho [5] have proposed a compact equation for Text Information Retrieval, which is modeled as a Bayesian Network. Their model is a similarity measure between a target-document and a query-document. Through a simplification process, which states that each Target Object is observed only if all terms \( k_i \in O_j \) and \( k_i \in O_m \forall i \), Berthier [16], Silva [19] and Coelho [5] achieved the following equation:

\[
P(O_j|O_m) = \eta \sum_k [1 - (1 - P(e1_j|k))] \times (1 - P(e2_j|k)) \times \ldots \times (1 - P(em_j|k)]
\]  

(6)

where \( P(e1_j|k) \) is the probability of occurrence of \( O_j \) as the term \( k \) for the evidences \( e1 \) was observed in the Model Object \( O_m \). Analogously, \( P(e2_j|k) \) is for evidence \( e2 \), and so far.

For this work, by considering the model of Fig. 2, we have adapted Equation (4) of Ribeiro-Neto [16], Silva [19] and Coelho [5], and present the following equation for CBIR:

\[
P(O_j|O_m) = 1 - ((1 - P(O_j|KC)) \times (1 - P(O_j|KF)))
\]  

(7)

Our proposed Bayesian Network differs from the Berthier [16] and Silva [19] model for text retrieval and from Coelho [5] model for image retrieval in several aspects. Firstly, our proposed approach does not use the vectorial model for computing the individual probabilities of evidences; we use, as we will see later, the shape and color evidences directly according to Equation (5); and the model proposed by Ribeiro-Neto [16], Silva [19] and Coelho [5] may attach weights for all evidences, as it can be seen in Equation (4).

Also, note that the belief network is used as a modeling framework and not as an inference engine. However, Equation (7) is the product of distinct pieces that represent the considered evidences; namely \( 1 - P(O_j|KC) \) and \( 1 - P(O_j|KF) \). In case we have only color evidences (for example, setting \( P(O_j|KF) = 0 \), this equation is reduced to \( P(O_j|O_m) = P(O_j|KC) \). The same occurs if we set \( P(O_j|KC) = 0 \); in this case, we have \( P(O_j|O_m) = P(O_j|KF) \).

Finally, the Bayesian network for two evidences, equivalent to the network in Fig. 2, is shown in Fig. 3.

In the case of color features, we have:

\[
P(O_j|KC) = \alpha \sum_{i=0}^{n} P(O_j|k_i^e)
\]  

(8)

Analogously, for shape features, we have:

\[
P(O_j|KF) = \beta \sum_{i=0}^{m} P(O_j|k_i^f)
\]  

(9)

In both equations, \( \alpha \) and \( \beta \) are normalization factors.

For the sake of explanation, in the remainder of this paper, we will refer to the \( P(O_j|KC) \) as color evidence, and \( P(O_j|KF) \) as shape evidence.

Figure 3. Bayesian network model concerning color and shape evidences

5 Experimental Results

We have proposed a method for object recognition based on a Bayesian Network for augmented reality applications. The first stage of our proposal consists in populating our network model with features of the objects of interest.
Then, by detecting facets in the image using segment detection, we extract the necessary features to make the correspondence between the 2D and 3D points (Fig. 4).

Once our system has been conceived to be developed on modules, we have some results using the ARToolkit pose estimation method [9]. Our first idea on the library was to isolate the pose estimation features based on markers to make it more general. As ARToolkit uses four coplanar points to calculate the position of the camera, the proposed system finds these points during tracking phase and use them as input to the ARToolkit module. This module returns the transformation matrix used for registration like in Fig. 5.

6 Conclusion

In this work we propose the use of the Bayesian Network Model to perform the initialization and the tracking phases of an Augmented Reality System. The main advantage of this model is the ability of record object features and support object recognition in the same architecture.

As our framework is not finished, we are researching other pose estimation methods [1, 23, 18] to be used in our application to accomplish the 3D registration process in real time. We are currently working on optimizations of our model to be able to use more complex objects and features.

References


