Non photorealistic simulation of video sequences for an accurate evaluation of tracking algorithms on complex scenes

Christine Dubreu, Antoine Manzanera, Eric Bohain

Non photorealistic simulation of video sequences for an accurate evaluation of tracking algorithms on complex scenes

Christine Dubreu[a], Antoine Manzanera[b], Eric Bohain[c]

[a] Cedip Infrared Systems and ENSTA, Paris, France;
[b] ENSTA, Paris, France;
[c] Cedip Infrared Systems, Paris, France

ABSTRACT

As target tracking is arousing more and more interest, the necessity to reliably assess tracking algorithms in any conditions is becoming essential. The evaluation of such algorithms requires a database of sequences representative of the whole range of conditions in which the tracking system is likely to operate, together with its associated ground truth. However, building such a database with real sequences, and collecting the associated ground truth appears to be hardly possible and very time-consuming.

Therefore, more and more often, synthetic sequences are generated by complex and heavy simulation platforms to evaluate the performance of tracking algorithms. Some methods have also been proposed using simple synthetic sequences generated without such complex simulation platforms. These sequences are generated from a finite number of discriminating parameters, and are statistically representative, as regards these parameters, of real sequences. They are very simple and not photorealistic.

This paper shows how reliable non-photorealistic synthetic sequences are, and how the number of parameters can be increased to synthesize more elaborated scenes in order to deal with more complex target and background texture characteristics and relative motion, including 3D deformations and occlusions. These synthesized sequences are easily generated from any desired scene characteristics, and can be reliably used for tracking algorithms evaluation in any conditions.

Keywords: Image processing, tracking, evaluation, simulation

1. INTRODUCTION

The success of automatic video surveillance of wide area scenes, which is currently arousing more and more interest, relies on the robustness of the tracking algorithms integrated to the systems. Thus, performance evaluation of such algorithms is becoming an increasingly important issue since it enables the developers to identify the weaknesses of their algorithms and to improve them.

For this purpose, we proposed an evaluation method,[1] in which tracking systems can be evaluated in any operational conditions (any target size, target to background contrast, velocity...) from a set of parameters describing theses conditions. The evaluation system is based on the generation of simple synthetic sequences comprising a moving target and a background. Synthetic sequences are generated from target and background texture models, dynamic deformations are applied. We want texture models and dynamic deformations to be fully characterized by a set of statistical parameters.

All the texture synthesis methods developed so far to generate textures required either a sample of the texture to be generated, or a huge list of statistical parameters from which a photorealistic texture could be obtained. In the evaluation method we proposed, we assumed that photorealism is not necessary for tracking

systems evaluation purpose, and proposed to synthesize textures from a small set of parameters defining the color, coarseness, directionality, regularity of a texture.

We also chose to model the deformations by geometric models. Therefore, only a small number of parameters defining the rotation, translation and scale between two consecutive frames is required.

We proposed an extremely simple texture synthesis method to generate a simple texture from a set of parameters, and modeled dynamics of the scene by geometric deformations, also fully characterized by a small set of parameters.

The synthetic scenes generated through this method were shown to be usable for the evaluation of tracking systems on simple single-target scenes. This method enables the generation of a large number of sequences on which tracking systems can be evaluated. The parameters on which the robustness of an algorithm depend are isolated, and sequences are generated with numerous values for these parameters, giving the minimum and maximum values of these parameters for which a tracking system would be robust. Figure 1 illustrates the performance of some tracking algorithms related to the value of a parameter.

Figure 1. Robustness of two tracking algorithms to the the target to background contrast change, and to the maximum target displacement and rotation between two frames

The aim of this paper is to define the reliability of this sequence generation method for the synthesis of more complex scenes aimed at tracking systems evaluation. Firstly, the non-photorealistic texture synthesis method performance for generating scenes that can be reliably used for algorithms evaluation are compared to the performance of other more complex and sophisticated methods, photorealistic, but requiring a texture sample.

Secondly, a way to improve the dynamic scene generation process once the synthetic textures generated is proposed, at the cost of additional input parameters. Therefore, more complex synthetic scenes can be generated which model for example occlusions and 3D deformations.

2. PRESENTATION OF THE THREE TEXTURE SYNTHESIS METHODS

This section overviews the three texture synthesis methods that will be used in the synthetic scene generation and for which the performance of tracking systems will be assessed.

2.1. Non photorealistic method

In the method we want to evaluate in this paper, operational criteria taken from a real scene are turned into objective measures and used to generate a synthetic dataset,[1] non-photorealistic, but statistically representative of the sequence that has to be simulated.

Indeed, the assumption is made that a minimal set of formal parameters can be defined, from which a synthetic scene likely to be used to evaluate tracking algorithms can be generated. This set is constituted of parameters defining the inherent statistical properties of the objects involved, i.e. the target and background (size, shape, histogram, texture characteristics such as coarseness, directionality, regularity...) and of parameters defining the relative interaction between these objects, and their temporal behavior (deformation, relative velocity, illumination changes...).

We concentrate on the parameters defining the inherent properties of objects to show that the minimal set we defined is sufficient for generating a scene reliably usable for our purpose. The obvious advantage of this method is that any value can be given to these operational criteria, enabling a tracking systems to

be evaluated in any conditions, in particular those for which no real scene nor texture sample is available. This enables developers to define and work on the weaknesses of their algorithms, and the users of systems to circumscribe the validity domain of the algorithms they use, and to choose the one that best fits the operational conditions.

2.2. Markov Random Fields based method

The first texture synthesis method used in comparison with the non-photorealistic one presented above is the algorithm described by Li-Yi Wei,[2] which requires a sample texture as input and generates textures with a very good perceived visual quality. The algorithm is derived from Markov Random Field texture models and generates textures through a sequential deterministic searching process.

It starts with an input texture sample S and a white random noise I. The random image I is forced to look like the sample by transforming it pixel by pixel in a raster scan ordering, i.e. from top to bottom and left to right. The value of each pixel p of I is determined using its spatial causal neighborhood $N(p)$, which is compared against all possible neighborhoods $N(p_i)$ from S, and p is assigned the value of the input pixel p_i with the most similar $N(p_i)$. This synthesis process ensures that the local similarity between I and S is maintained. The figure 2 shows the results of this texture synthesis method from samples taken from the Brodatz[3] database. These texture samples have a high regularity, and the synthesized textures are photorealistic, since they are visually similar to the texture sample, and the regularity is kept.

2.3. Third method : Wavelet-based method

The second texture synthesis method to be compared to the non-photorealistic one is the texture synthesis method developed by Portilla & Simoncelli[4] . It is also based on the establishment of a set of statistical measurements such that two textures would be identical in appearance if and only if they agree on these measurements. These statistical measurements are extracted from a sample, and they are much more numerous since photorealism is taken into consideration.

A texture sample is first decomposed using a multi-scale oriented linear basis, from which complex coefficients are computed, as well as statistical moments of the pixel distribution, such as mean, variance, range... The number of parameters depends on the number of subbands and the size of spatial neighborhoods used. In the examples given in this paper, a total of 710 parameters is used, but satisfactory results can usually be obtained with a small subset of these parameters, depending on the texture to analyze.

From an image of Gaussian white noise, the sample statistics of each subband of a pyramid are forced to match those of a reference image, then the image is reconstructed, and the statistics of the resulting pixels are forced to match those of the texture sample. This process is iterated several times, until convergence of the image. More details may be found in the references given below.[5] The figure 2 also shows the results of this texture synthesis method on some texture samples extracted from the Brodatz database after 25 iterations. Again, on these highly regular samples, the regularity is kept, and the synthetic texture is visually similar to the sample one.

Figure 2. Textures synthesized using the Markov-Based Method (second line), and using the Wavelet-Based Method (third line), from patchs from the Brodatz texture database (first line)

3. EVALUATION OF THE THREE METHODS

3.1. Aim

The aim of this section is to compare the reliability of each of the three texture synthesis methods of the whole synthetic scene generation process for tracking systems evaluation.

3.2. Evaluation protocol

A large number of video sequences representative of the range of operating conditions of a tracking system are taken. These sequences include some infrared and visible real sequences, taken from naval, airborne, and ground cameras, for which a ground truth has been manually generated, and the sequences extracted from the VIVID database,[6] which is a database of infrared and visible video sequences together with their ground truth aimed at evaluation of tracking systems. Samples of the target and background textures are extracted for each of these sequences. From these samples, synthetic sequences are generated using the three texture synthesis methods described in section 1, and the target is given a motion and deformation similar to the real one provided by the ground truth of the real scenes.

Only one simple-textured target is synthesized, since the aim of this comparison is to validate the assumption that photorealism is not necessary to evaluate low-level object tracking algorithms, but some multi-target sequences could be generated from several samples.

Tracking algorithms are then tested on each set of four video sequences (one real, two photorealistic, and one non-photorealistic). The performance of the tracking algorithms on each of these sequences are compared to see how reliably the synthetic ones can be used instead of the real ones for an accurate evaluation of the systems.

3.3. Algorithms

Two algorithms are used for this evaluation process: a correlation algorithm, and a centroid algorithm. The correlation algorithm relies on the use of a reference image A representing the target, and the search of the position for which the correlation value between A and a rectangular patch B in the current frame, given by (1), is maximal. The search strategy is based on a multiresolution gradient descent with a diamond pattern.[7]

$$r = \frac{\sum_i \sum_j (A_{ij} - \bar{A})(B_{ij} - \bar{B})}{\sqrt{\left(\sum_i \sum_j (A_{ij} - \bar{A})^2\right)\left(\sum_i \sum_j (B_{ij} - \bar{B})^2\right)}} \tag{1}$$

The second algorithm is the centroid algorithm described by Albus & al,[8] in which a probability map is used to determine whether pixels belong or not to the target, and to determine the target's center of gravity. This algorithm uses concentric gates to determine relevant regions: the outer region, which contains mostly background pixels, and the inner region mostly target pixels. Then, a probability map can be computed from the smoothed histograms of these regions to segment the target :

$$P(k) = \frac{H_S^I(k)}{H_S^I(k) + H_S^O(k)}, \forall k \in 0...\Delta - 1, \tag{2}$$

where H_S^I and H_S^O are respectively the smoothed histogram of the inner and outer region, and Δ is the number of grey levels in the image. $P(k)$ is the probability for a pixel of intensity k to belong to the target.

3.4. Metric

A metric has to be defined for the evaluation of these tracking algorithms. A diverse range of measures and procedures to establish a performance metric has been used in tracking evaluation.[9] The choice depends on the target application, as the priorities will vary for different applications. Two metrics are used in this paper :

$$r_1 = \sum_{i=1}^{n} \frac{f(i)}{n}$$

$$r_2 = \frac{\sum_{i=1}^{n} \sigma * f(i)}{\sum_{i=1}^{n} f(i)}$$

, where σ is the standard deviation of the error between the real position of the target center given by the ground truth and the position given by the algorithms all over the sequence, i is the index of the video frame, n the total number of video frames of the sequence, and f a function determining whether the target is believed to be found or not :

$$f(i) = \begin{cases} 1 \text{ if the target is believed to be found at frame } i \\ 0 \text{ if the target is believed to be lost at frame } i \end{cases}$$

3.5. Results

The figure 3 shows how the three texture synthesis algorithms perform to synthesize a texture image from samples of complex, often non regular and non-stationnary- textures, or from some statistical measurements taken from these samples.

Real sample	Non photorealistic synthesized texture	Markov-based synthesized texture	Wavelet-based synthesized texture

Figure 3. Examples of texture synthesized from target and background samples extracted from real sequences

The table 1 shows the performance of the two tracking algorithms on real and synthetic sequences for three sets of sequences representative of all the sequences used for this evaluation. The sequences used here for the evaluation of the performance of the algorithms are more complex that the sequence that were used for the validation of this method for evaluation of low-level algorithms on simple scenes. Indeed, in these sequences, the target and background textures are more complex, since they can be non-stationary, or completely irregular...

It can be noticed, in the case of these more complex scenes, that the scene generation method using the non-photorealistic texture synthesis does not perform as well as methods using the photorealistic ones. This shows that, although this texture synthesis method is efficient for evaluation of tracking algorithms on scenes where target and background are simple-textured, it has its limits, and is no longer robust when the texture of the target or background becomes non-regular, or non-stationary. Therefore, it is preferable to use the photorealistic ones in these conditions, provided that texture samples are available.

Algorithm	Real Scene	Non photorealistic synthetic scene	Synthetic scene generated by Neighborhood Searching	Synthetic Scene generated by the wavelet-based method
Correlation	1.71	1.43	1.52	1.67
Centroid	2.43	0.97	1.04	1.52
Correlation	4.25	3.23	3.76	3.56
Centroid	6.32	3.59	5.38	4.4
Correlation	0.54	0.44	0.43	0.51
Centroid	1.84	1.47	1.84	1.64

Table 1. Standard deviations of the error between the position of the target given by the ground truth and by the correlation and centroid algorithms on several sets of sequences for which the target and background texture parameters are equal.

Besides, these scene generation methods are not robust when evaluating tracking algorithms on scenes with occlusions, or three-dimensional target deformations. Indeed, the very simple geometric transformation model of the motion and deformations is aimed at evaluating the performance of low level algorithms, i.e. the performance of algorithms on single-target basic sequences. Therefore, this model has to be improved by adding new parameters, characterizing the 3D deformations, and the occlusions.

4. IMPROVEMENT OF THE DYNAMIC GENERATION METHOD

In this section,

4.1. Basic dynamic motion modeling

Once the target(s) and background texture fields are created using this method, the target(s) field(s) are mapped on an ellipse with eccentricity and size corresponding to the input shape parameters, and superimposed to the background field. A synthetic dynamic scene is then computed in which a dynamic motion characterized by the displacement parameters defined in 2.1is given to the target and background.

A 2D deformation is applied to the target at each frame generation. This deformation is modeled by a composition of a rotation, a translation, and a scale :

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \rho \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} T_{\mathrm{X}} \\ T_{\mathrm{Y}} \end{pmatrix} \tag{3}$$

Therefore, the parameters $\theta^{\mathrm{T}}_{\mathrm{max}}$, $\rho^{\mathrm{T}}_{\mathrm{max}}$, and $\mathrm{T}^{\mathrm{T}}_{\mathrm{Xmax}}$ and $\mathrm{T}^{\mathrm{T}}_{\mathrm{Ymax}}$, defining respectively the maximum rotational angle, scale, and displacement of the target(s) between two consecutive frames have to be defined. Similarly, the tbackground's motion is defined by the parameters $\mathrm{T}^{\mathrm{B}}_{\mathrm{Xmax}}$ and $\mathrm{T}^{\mathrm{B}}_{\mathrm{Ymax}}$, and its deformation by $\theta^{\mathrm{B}}_{\mathrm{max}}$, and $\rho^{\mathrm{B}}_{\mathrm{max}}$.

To avoid discontinuities and re-synthesis of new pixels at each frame generation, the background texture is previously symmetrized and infinitely replicated in the x and y directions. The deformation and displacement of the background and target are generated in such a way that it is continuous; there is no sudden change in the angle of rotation, scale, or displacement vector, which is consistent with what is commonly found in real sequences.

4.2. Modeling of the 3D deformations

The displacement and deformation of the objects present in the scene were chosen to be modeled in two dimensions, since it is a good approximation of the scene dynamics between two frames. However, modeling the 3D motion and deformations of the target and background in a scene would give sequences with dynamics closer to the reality and would give better results on the performance evaluation of tracking systems. Such deformations can be modeled by a non-stationarity of the texture parameters (histogram, coarseness, directionality) : a temporal parameter is added, on which the texture parameters are indexed.

For example, a maximal variation of the histogram can be allowed between two consecutive image frames, and a value given to a temporal parameter, controlling the change in the histogram. A transformation depending on the value of the temporal parameter is then applied to the histogram, resulting in it being more or less spread. This very simple process of histogram modification is shown in figure 4 and shows how illumination changes can be modeled.

Figure 4. Example of texture histogram modification between two frames

Similarly,the coarseness and directionality of the texture can be indexed to a temporal parameter : between two consecutive frames, a scale can be applied to the texture, which coefficients are indexed to a temporal parameter, resulting in a coarser or finer texture.

The addition of a temporal parameter to which each of the texture parameter of the objects in the scene is indexed enables system designers to get more precise specifications on the validity domain of their algorithms, and to choose efficiently the best algorithm to use for a given application.

4.3. Modeling of the occlusions

The dynamics of the synthetic scene can also be improved by the modeling of static and dynamic occlusions, as they often occur in real scenes.

Static occlusions can be modeled by considering a discrete number of plans, and by adding a depth parameter to every point of the target or background. The probability of occlusions is linked to the number of background pixels having a smaller depth than the target. Figure 5 shows the results of a scene generation with the same textural properties, but with different probabilities of static occlusions.

$$p = 0 \quad p = 0.2 \quad p = 0.4 \quad p = 0.7$$

Figure 5. Example of occlusions modeling, together with the probabilities of occurrence of these occlusions.

Dynamic occlusions can be modeled in the same way, by synthesizing a sequence with multiple targets, and by adding to each pixel of each target and background a depth parameter. The superimposition of all isolated targets allows the generation of a large number of sequences representing different scenarii. The duration and occurrence of the occlusions can be controlled by the motion of each of the isolated targets.[10]

5. CONCLUSION AND FURTHER WORK

We have shown in this paper that the use of non-photorealistic texture synthesis in the generation of synthetic scenes for tracking system evaluation is relevant. Indeed, the performance of tracking systems on real sequences are comparable to their performance on synthetic sequences, whether they are photorealistic or not. therefore, the non photorealistic method provides us with an efficient way to quantitatively evaluate low-level object tracking methods and get their validity domain without having to use costly and heavy simulation platforms.

In order to improve high level tracking algorithms evaluation, we propose a way to model 3D deformations, and static and dynamic occlusions, and more complex target(s) and background textures. This results in an increase in the number of dynamic discriminating parameters, and enables us to get more accurate circumscription of the validity domain of tracking algorithms.

REFERENCES

1. C. Dubreu, A. Manzanera, E. Bohain : "Comprehensive Evaluation of Tracking Systems by Non-Photorealistic Simulation", SPIE, Acquisition, Tracking, and Pointing, April 2007.

2. L-Y. Wei : "Texture Synthesis by Fixed Neighborhood Searching", Ph.D dissertation, Stanford University, November 2001.

3. P. Brodatz : "Textures : A photographic album for Artists and Designers", Dover, New York, 1966.
4. J. Portilla, E. P. Simoncelli : "A Parametric Texture Model Based on Joint Statistics of Complex Wavelet Coefficients", int Journal of Computer Vision. 40(1):49-71, October 2000.

5. J. Portilla, E. P. Simoncelli : "Texture Modeling and Synthesis using Joint Statistics of Complex Wavelet Coefficients",IEEE Workshop on Statistical and Computational Theories of Vision, June 1999.

6. R. T. Collins, X. Zhou, and S. K. Teh : "An Open Source Tracking Testbed and Evaluation Web Site", IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS 2005), January, 2005.

7. A. Barjatya : "Block matching algorithms for motion estimation", 2004.

8. J. E. Albus, L. J. Lewins, J. R. Schacht : "Centroid Tracking using a probability map for target segmentation" SPIE, Acquisition, Tracking, and Pointing, April 2002.

9. T. Ellis : "Performance Metrics and Methods for Tracking in Surveillance", Proceedings of the third IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS 02), pp 26-31, September 2000.

10. J. Black, T. Ellis, P. Rosin : "A Novel Method for Video Tracking Performance Evaluation", Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (PETS), January 2005.