Trajectory Similarity Measures Using Minimal Paths

Brais Cancela, Marcos Ortega, Alba Fernández, and Manuel G. Penedo

University of A Coruña,
Varpa Group, Department of Computer Science,
Campus de Elviña, s/n, Spain
{brais.cancela,mortega,alba.fernandez,mgpenedo}@udc.es
http://www.varpa.org

Abstract. Dealing with surveillance systems, large amount of distance measures are presented in order to classify both normal and abnormal behavior. Typically, techniques based in point-to-point distances are used. However, these techniques do not take into account information about the environment, like pits or restricted areas, for instance. Using a minimal path algorithm to model the usual paths, we develop new trajectory distance measures that are able to introduce information about the scene. The results obtained show promising results.

1 Introduction

Detecting human activities and behavior is a huge field of study in computer vision. One of the most active topics is related with the study of human behavior and their group relationships. This field has a special interest in surveillance systems. The idea of being able to detect abnormal behavior has being widely study. For instance, a strange movement could result in an abnormal behavior which has to be detected in order to throw an alarm.

The classical path classification methodologies are based in clustering techniques. Different configurations were used: direct [15], using techniques like k-means or fuzzy c means; agglomerative [5], where we merge clusters until we obtain the desired number; divisive [4], the top-down dual to agglomerative clustering; Hybrid [11], Graph-based [14] or Spectral [10]. All the techniques mentioned above are limited, since they require routes with the same number of samples to compute the clusters, and they are not easy to update along time. Suppose, for example, that an usual target is interrupted because of an object placed in the track. A new cluster is created with the new routes, but the previous cluster still remains in the system. A target which decides to jump that object, which clearly is an abnormal movement, will be declared as normal behavior because of the existing cluster. More recent techniques include the use of nonparametric Bayesian models [20], [19] or use models to predict the motion behavior [8].

To overcome the clustering issue, we developed a new system that can be easily updated [6]. We use a minimal path algorithm to model the abnormal behavior. In normal situations, a target tends to choose the route that costs the least time to reach its desired destination. Thus, trajectories that highly differ from this "ideal" route are marked as abnormal. A new metric were presented, based in a distance map algorithm, which requires a high computational time.

A. Petrosino (Ed.): ICIAP 2013, Part I, LNCS 8156, pp. 400–409, 2013.

In this work we present a new trajectory distance measure that can be used in surveillance systems to detect abnormal behavior. This technique uses the properties of the minimal path algorithm to obtain a metric without increasing the computational time, solving the distance map technique disadvantage. This paper is organized as follows: section 2 shows different trajectory measure algorithms; section 3 describes our method; section 4 shows some experimental results and section 5 offers conclusions and future work.

2 State of the Art

We define a trajectory as a collection of the positions a target reach along its way. So, we have a collection of N positions defining a target route. To compare it, there exist in the literature different approaches for trajectory distance metrics. Methods based in classical measure techniques were used, like euclidean distance [10] or principal component analysis (PCA) [3]. However, these methods obtain poor results, requiring trajectories with the same size to be compared. Other attempts, like the modified Hausdorff distance [2], does not take into account the order into the trajectory points.

Thus, distance measure techniques have to be able to compare unequal length trajectories, while taking into account the route orientation. In [12], Keogh et al. presented the Dynamic Time Warping (DTW) technique. Basically, this method tries to find a time warping that minimizes the distance between two different trajectories. It can be used with trajectories with different sizes. Buzan et al. [5] introduced a similar idea, the Longest Common Subsequence (LCSS). It can also be used with unequal length data, becoming more robust to noise. The reason is that not all the trajectory points need to be matched. Similar to these methods, Piciarelli and Foresti (PF) [17] uses a dynamic time warping window, which is increased along time, that is, the maximum error allowed is low at the starting trajectory point, becoming larger while we are reaching the end. The performance of these metrics were tested in [16].

Although these methods can deal with the problems mentioned before, they all present a major issue for real domains: they do not take into account information about the environment. For instance, in Fig. 1, we can see two examples of situations these techniques cannot correctly address. In the left, two parallel trajectories are defined. Using the similarity measure it is easy to conclude that both trajectories are similar. However, the red route is produced by a counterclockwise car, which clearly is an abnormal behavior. On the right image, both the red and the green route are similar to the blue one, but the red one crosses the central reservation.

To solve situations like the previously illustrated one, we develop a methodology that is able to introduce information about the environment [6], based in the geodesic active contours [7] and the level set theory [13]. A modification of the minimal path approach using geodesic active contours performed by Cohen and Kimmel [9] is provided. In this work, starting at any given point p_0, a minimal path map over an image is obtained, with a $\mathcal{O}(N \log N)$ complexity, being N the number of pixels in the image. To do that, a potential image P is created, which includes information about the environment. Later, the potential used in the algorithm is defined as

$$\tilde{P}(p) = \omega + P(p), \tag{1}$$

(a) (b)

Fig. 1. These trajectories are defined as similar using classical distance measure techniques, while including the scene information the red routes are clearly abnormal

being ω the regularization term. This potential has to be defined so $\tilde{P} > 0$ and $P \geq 0$, meaning that $\omega > 0$. Values close to 0 implies pixels that are easy to reach, while higher values means the opposite.

Having this potential, the surface of minimal action U can be created. This surface assigns a value to each pixel in the image, which corresponds with the least cost that takes to reach that pixel, starting at the initial point p_0. To create that surface, the Fast Marching Method (FMM) proposed by Sethian is used [18]. An ordered upwind scheme is used, updating the cost of a given pixel $U_{i,j} = u$ following the equation

$$(\max\{u - U_{i-1,j}, u + U_{i+1,j}, 0\})^2 + (\max\{u - U_{i,j-1}, u + U_{i,j+1}, 0\})^2 = \tilde{P}^2_{i,j}. \quad (2)$$

Once the minimal path between the initial and the final point is obtained, we can evaluate the real path against this minimal cost approach. Note that the final point mentioned before is not necessarily the point where we stop to track a target. The minimal path can be computed each frame a target is detected, using the last position detected as final path. This is an advantage with respect to other methodologies, since the abnormal behavior can be detected as soon as it occurs.

So, in order to evaluate the real path against the minimal path approach, a distance map image is created. Having this minimal path algorithm in mind, it is easy to see that, if using as initial points all the points in a given trajectory $C(s)$ instead of the point p_0 in the original algorithm, the surface of minimal action U we obtain is a distance map, where the value $U_{i,j}$ of each pixel is the minimum cost that takes to reach that point, starting at any point that belongs to the trajectory $C(s)$. Note that, the distance between two consecutive pixels is defined by the potential $\tilde{P}$. In [6], a discussion about the potential image is provided.

The problem of this method is the computational cost needed to create the distance image. Since the algorithm is, in essence, equivalent to the minimal path approach, the cost to obtain the map is $\mathcal{O}(N \log N)$. Thus, our goal is to create a new trajectory distance measure that is able to include information about the environment, taking into account the properties of the minimal path approach, but avoiding unnecessary extra computations.

3 Minimal Path Metrics

Our goal is to use the properties of the Fast Marching Method used to create the minimal action surface U to detect if a trajectory is abnormal. That surface of minimal action has a convex like behavior, in the sense that, starting at any given point p, and following the gradient descent direction in U, we always converge to the initial point p_0. This means that the minimal action surface U has one local minimum, which is $U(p_0) = 0$. Furthermore, this geodesic active contour-based technique is consistent with the continuous problem, in the sense that the solution provided by the FMM becomes closer to the exact solution while reducing the grid. This property allows this algorithm to avoid the 'metrication error', which appears in the classical graph search algorithms, like A* or F*.

This property is crucial to present our methodology. If, for instance, we introduce the potential image $\tilde{P} = \tau > 0$, and we compute the minimal action surface U, starting at any given point within the grid p_0, we obtain a distance map, as the value at any point p, U_0, is the distance between this point with respect to the initial one p_0. And, contrary to the graph search algorithms, since the FMM is consistent with the continuous case, the solution we obtain is close enough to the euclidean distance. Note that this system is isotropic, having no information about directional forces. The cost to reach a point is always the same. no matter which direction the front-propagation come by.

However, using a potential like the mentioned before causes the system to lose the information about the environment, since this potential is constant all over the space. Fortunately, it is possible to compute the distance map regardless the type of potential used. We have to define another minimal action surface D, which is going to be updated using the equation

$$D(p) = \begin{cases} \dfrac{D(p_a)+D(p_b)+\sqrt{2\tau^2-(D(p_b)-D(p_a))^2}}{2} & \text{if } \tilde{P}(p) > (U(p_b)-U(p_a)) \\ D(p_a)+\tau & \text{otherwise} \end{cases}, \tag{3}$$

where p_a and p_b are the neighbors used in the Eq. 2 to update the surface of minimal action $U(p)$. satisfying that $U(p_a) \leq U(p_b)$, and τ is the distance between two neighbor pixels. Typically, $\tau = 1$.

Thus, while computing the minimal action surface U we can compute the distance map D without any substantial cost increment. In algorithm 1 a pseudocode of our FMM implementation is presented.

Once we have the distance map related to a trajectory, we are able to perform a similarity measure that can detect abnormal behavior. For notation, we have a trajectory

$$Tr = \{p_0, p_1, \ldots, p_M\}, \tag{4}$$

where each point represents positions that are reached for the target. Note that this is an ordered sequence of events, where the position p_i is reached before the position p_{i+1}. These trajectories can be obtained using tracking techniques, being p_0 the first time a target is tracked. Since, as we demonstrate in [6], a target usually tends to follow the

Algorithm 1. Distance Surface Method

Definitions:

- *Alive* set: points of the grid for which U has been computed and it will not be modified.
- *Trial* set: next points in the grid to be examined (4-connectivity) for which an estimation of U is computed using the points in *alive* set.
- *Far* set: the remaining points of the grid for which there is not an estimate for U.

Initialization:

- For each point in the grid, let $U_{i,j} = \infty$ (large positive value).
  Put all points in the *far* set.
- Set the initial point p_0 to be zero:
  $U_{p_0} = 0, D_{p_0} = 0$, and put it in the *trial* set.

Marching loop:

- Select $p = (i_{min}, j_{min})$ from *trial* with the lowest value of U.
- If p is equal to p_1 being p_1 the final point then we finish.
- Else put p in *alive* and remove it from the *trial* set.
- If $\tilde{P}(i_{min}, j_{min}) < \tau$, for each of the 4 neighboring grid points (k, l) of (i_{min}, j_{min}):
  - If (k, l) belongs to *far* set, then put (k, l) in *trial* set.
  - If (k, l) is not in *alive* set, then set $U_{k,l}$ with Equation 2
  - and set $D_{k,l}$ with Equation 3.

path that cost less effort to reach the goal, we can conclude the relation between the minimal path distance with respect to a normal trajectory behavior is close to 1, that is,

$$\frac{\sum_{i=2}^{M} d(p_{i-1}, p_i)}{D(p_M)} \approx 1, \tag{5}$$

where $d(p_{i-1}, p_i)$ is the distance between two consecutive points in the trajectory. To compute this distance it is also possible to use the minimal path approach, starting in p_{i-1} instead of p_1. However, the points contained in the route are usually close together, meaning the euclidean distance often results in a good approach.

Having this in mind, different metrics are presented for detecting abnormal behavior. All these metrics are based in two different equations. The first one tries to obtain the relation between the target route and its associated minimal path. We called it Minpath Relation (MR), and is defined by

$$MR(p_N) = \left(\frac{\sum_{i=2}^{N} d(p_{i-1}, p_i)}{D(p_N)} - 1 \right)^2, \tag{6}$$

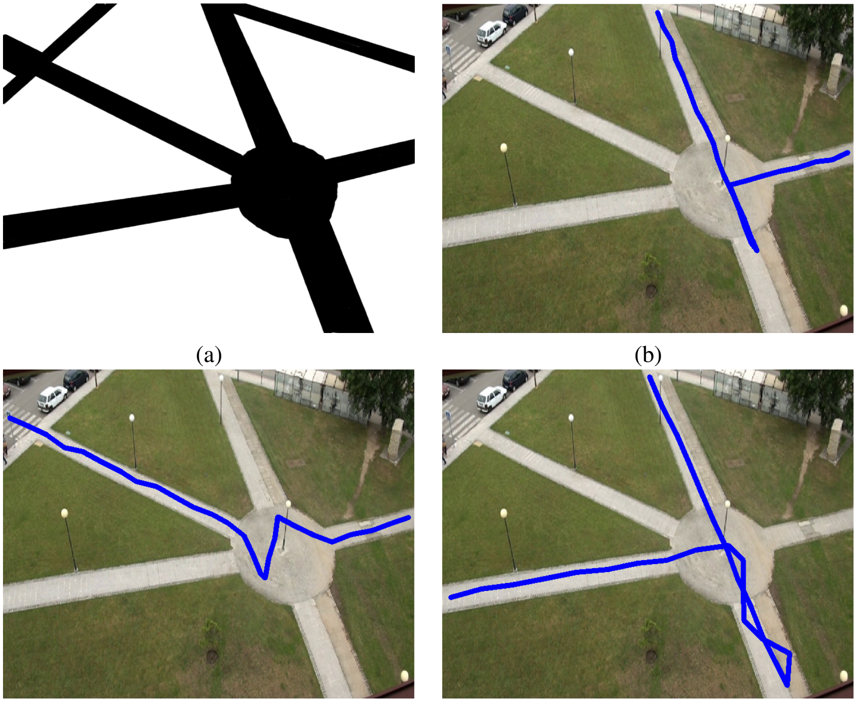
where $1 < N \leq M$. The second one tries to detect local variations in the MR metric. We called it Local Minpath Relation (LMR), and is defined by

$$LMR(p_N) = \left(\frac{d(p_{N-1}, p_N)}{D(p_N) - D(p_{N-1})} - 1 \right)^2. \tag{7}$$

In both metrics, values close to 0 mean the path is correct, while higher values could indicate an abnormal behavior.

4 Experimental Results

In our experiments we have used a dataset publicly available, BARD [1], in order to test the methodology. Several trajectories are developed over an intersection scene, resulting in more than 15000 trajectory points. In the experiments we decided to use a fixed potential image, that can be shown in Fig. 2-(a). We consider the grass areas as forbidden areas, using high values in the potential to model them.



(a) (b) (c) (d)

Fig. 2. Path trajectory examples. (a) Potential image used to computed the minimal path. (b, c, d) Abnormal behavior examples.

In first place, we decided to test different approaches of the metrics explained before. We decided to test, according to the MR equation, both MR, the Mean MR

$$MMR(p_N) = \frac{\sum_{i=2}^{N} MR(p_i)}{N}, \tag{8}$$

its variance

$$VMR(p_N) = \frac{\sum_{i=2}^{N} MR(p_i)^2}{N} - MMR(p_N)^2 \quad (9)$$

and its top value

$$TMR(p_N) = \max_{i \in [2..N]} p_N. \quad (10)$$

When dealing with the LMR equation, the mean (MLMR), the variance (VLMR) and the maximum (TLMR) are used. As mentioned earlier, values close to 0 mean there is no abnormal behavior in the route. Thus, we can detect abnormal situations by simply introducing a threshold. In Table 1 we show the results obtained using these metric over the BARD dataset. Looking at the ROC Area Under the Curve values obtained, we can conclude the metrics exposed obtain good results, especially MMR and TMR measures. Most of the errors achieved by these metrics are related with the difficulty of annotate the correct moment where a normal route starts being erratic. As a result, it is possible that some errors may occur because of bad manually annotations.

Table 1. Minpath Metric Results. Using the ROC Area Under the Curve metric, we found that all the techniques achieve good results.

Metric	**ROC AUC**
MR	0.9417
MMR	0.9691
VMR	0.9578
TMR	0.9673
MLMR	0.9535
VLMR	0.9448
TLMR	0.9512

Having these results, we conclude the MMR technique achieve the better results. However, this comparison is made by using a potential image with the same size of the video frame, in this case, (576×720). Although the computation of these metrics is equivalent to the computation of the minimal path method, $\mathcal{O}(N \log N)$, the time needed is too high to be used in real-time systems. This is not a problem when dealing with a fixed potential, where the potential cannot be updated along frames, because we only need to compute the minimal path one time per each target. Storing the minimal action surface U, we can obtain the MMR metric with a $\mathcal{O}(1)$ complexity.

However, in more complex scenarios, we need to use a potential that is going to be modified along time. In this case, we need to compute the minimal action surface U every time we want to obtain the MMR metric. Thus, we need to reduce the potential image size in order to reduce the computational time. In Fig. 2 we can see the results obtained by reducing the image. As we can see, similar results are obtained if we reduce the potential image to (72×90), allowing our method to speed up the response without decreasing significantly the performance of the metric.

In order to compare the results of the metrics mentioned before against the baseline techniques, we use the ROC curve. In Fig. 3 we compare our metrics with the baseline

Table 2. Minpath Metric Results. Using the ROC Area Under the Curve metric, we found that all the techniques achieve good results.

Decreasing factor	ROC AUC MR	MMR	VMR	TMR	LMR	VLMR	TLMR
f = 1 (576×720)	0.9417	**0.9691**	0.9578	0.9673	0.9535	0.9448	0.9512
f = 2 (288×360)	0.9488	**0.9646**	0.9519	0.9619	0.9493	0.9411	0.9471
f = 4 (144×180)	0.9574	**0.9581**	0.9424	0.9545	0.9453	0.9379	0.9438
f = 8 (72×90)	**0.9528**	0.9480	0.9224	0.9395	0.9308	0.9267	0.9326
f = 16 (36×45)	0.8965	**0.9124**	0.8678	0.8909	0.8646	0.8695	0.8712
f = 32 (18×22)	0.7912	**0.8598**	0.8179	0.8371	0.7214	0.7290	0.7327

methods. Since our method clearly outperforms methods that need samples with the same size to perform the computation, we decide to compare our method against more powerful techniques, like the previously mentioned PF, LCSS and DTW. Moreover, we introduce our previous distance map based techniques. We can conclude that the MMR metric clearly outperforms the baseline methods, except the Weighted Distance Map. However, the MMR can obtain similar results while avoiding the computation of the distance map image, which has a $\mathcal{O}(N \log N)$ complexity, meaning our new method is more suitable for being used in real-time applications.

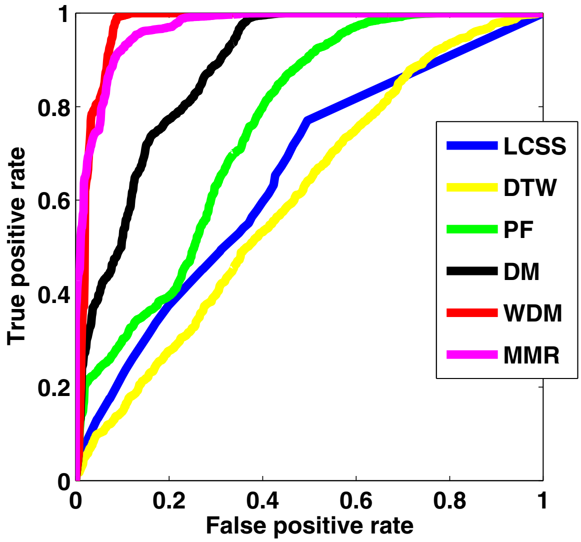


Fig. 3. ROC curve. Our new metric outperforms the baseline methods, with the exception of the Weighted Distance Map. However, the computational cost allows our new method to be more suitable to be used in real-time environments.

5 Conclusions

In this paper we present new metrics to detect abnormal behavior using target trajectories. Using minimal path techniques, which are proven to be useful detecting abnormal behavior, we develop new metrics that make use of the surface of minimal action properties, without increasing significantly the computation of such minimal paths. A comparison between different new metrics is performed, where the MMR obtains the better results. The results also show the potential image can be substantially reduced, having no significantly yield loss. Another comparison against baseline methods demonstrates our methods outperform classical abnormal behavior metrics, while obtains similar results to the recent Weighted Distance Map metric, which is proven to be computationally expensive.

In future works, we plan to introduce both direction and speed in our minimal path algorithm, allowing our system to have more information, in order to detect abnormal behavior that is not provided in our recent algorithm, like sudden speed changes or movements in the opposite direction of the usual routes.

Acknowledgments. This paper has been partly funded by the Consellería de Industria. Xunta de Galicia through grant contract 10TIC009CT, and by the Ministerio de Ciencia e Innovación through grant contract TIN2011-25476.

References

1. BARD, behavioral analysis and recognition dataset, `http://www.varpa.org/bard/`
2. Atev, S., Masoud, O., Papanikolopoulos, N.: Learning traffic patterns at intersections by spectral clustering of motion trajectories. In: 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4851–4856. IEEE (2006)
3. Bashir, F., Khokhar, A., Schonfeld, D.: Object trajectory-based activity classification and recognition using hidden markov models. IEEE Transactions on Image Processing 16(7), 1912–1919 (2007)
4. Biliotti, D., Antonini, G., Thiran, J.P.: Multi-layer hierarchical clustering of pedestrian trajectories for automatic counting of people in video sequences. In: Proceedings of the IEEE Workshop on Motion and Video Computing (WACV/MOTION 2005), vol. 2, pp. 50–57. IEEE Computer Society, Washington, DC (2005)
5. Buzan, D., Sclaroff, S., Kollios, G.: Extraction and clustering of motion trajectories in video. In: Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004, vol. 2, pp. 521–524. IEEE (2004)
6. Cancela, B., Ortega, M., Penedo, M., Novo, J., Barreira, N.: On the use of a minimal path approach for target trajectory analysis. Pattern Recognition 46(7), 2015–2027 (2013)
7. Caselles, V., Kimmel, R., Sapiro, G.: Geodesic active contours. International Journal of Computer Vision 22(1), 61–79 (1997)
8. Chen, Z., Wang, L., Yung, N.H.: Adaptive human motion analysis and prediction. Pattern Recognition 44(12), 2902–2914 (2011)
9. Cohen, L., Kimmel, R.: Global minimum for active contour models: A minimal path approach. International Journal of Computer Vision 24(1), 57–78 (1997)
10. Hu, W., Xie, D., Fu, Z., Zeng, W., Maybank, S.: Semantic-based surveillance video retrieval. IEEE Transactions on Image Processing 16(4), 1168–1181 (2007)

11. Karypis, G., Han, E.H., Kumar, V.: Chameleon: Hierarchical clustering using dynamic modeling. Computer 32(8), 68–75 (1999)
12. Keogh, E., Pazzani, M.: Scaling up dynamic time warping for datamining applications. In: Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 285–289. ACM (2000)
13. Kimmel, R., Amir, A., Bruckstein, A.: Finding shortest paths on surfaces using level sets propagation. IEEE Transactions on Pattern Analysis and Machine Intelligence 17(6), 635–640 (1995)
14. Li, X., Hu, W., Hu, W.: A coarse-to-fine strategy for vehicle motion trajectory clustering. In: The 18th International Conference on Pattern Recognition, ICPR 2006, vol. 1, pp. 591–594 (2006)
15. Morris, B., Trivedi, M.M.: An adaptive scene description for activity analysis in surveillance video. In: ICPR, pp. 1–4. IEEE (2008)
16. Morris, B., Trivedi, M.M.: Learning trajectory patterns by clustering: Experimental studies and comparative evaluation. In: CVPR, pp. 312–319. IEEE (2009)
17. Piciarelli, C., Foresti, G.: On-line trajectory clustering for anomalous events detection. Pattern Recognition Letters 27(15), 1835–1842 (2006)
18. Sethian, J.: A fast marching level set method for monotonically advancing fronts. Proceedings of the National Academy of Sciences 93(4), 1591–1595 (1996)
19. Wang, X., Ma, K.T., Ng, G.W., Grimson, W.E.L.: Trajectory analysis and semantic region modeling using nonparametric hierarchical bayesian models. International Journal of Computer Vision 95(3), 287–312 (2011)
20. Wang, X., Ma, X., Grimson, W.E.L.: Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models. IEEE Transactions on Pattern Analysis and Machine Intelligence 31(3), 539–555 (2009)