Multi-camera indoor video processing for context awareness

L. Marcenaro
TechnoAware srl
Genova, Italy
lucio.marcenaro@taw.it

I. Magliano
DIBE University of Genoa
Genova, Italy
ingrid@dibe.unige.it

A. Beoldo
DIBE University of Genoa
Genova, Italy
beoldo@dibe.unige.it

M. Valla
Telecom Italia
Torino, Italy

C.S. Regazzoni
DIBE University of Genoa
Genova, Italy
massimo.valla@telecomitalia.itcarlo@dibe.unige.it

ABSTRACT

In this paper a system is presented able to acquire images from multiple indoor network cameras and extract contextual information about persons detected within the considered environment. Distributed system architecture allows one to process images from several cameras on a network of PCs. Objects tracking and posture classification techniques are used in order to extract contextual information from video. These information are stored in a remote database that is accessed from an higher level application that is able to interact with users' mobile phones for delivering context-based-services. In particular, proposed scene understanding techniques have been used for implementing an automatic terminal silencing service in case of a meeting and a sos-call in case of a falling person.

Keywords

ambient intelligence context aware applications, video surveillance, video processing

1. INTRODUCTION

Ambient awareness can be defined as the process of acquiring, processing and acting upon contextual information: a system is "context aware" if it can respond to certain situations or stimuli in its environment. Main issues of a context awareness system are [12]:

- Contextual sensing: sensing is the most basic part of context awareness. A sensing device detects various environmental states, like position, time, and presents them to the user or to the services that want to make use of them. This sensing is not restricted to reading out hardware sensors; it is equally applicable for synthesising context information, for instance using place and time to determine if it is dark outside.
- Contextual adaptation: using context information, services can adapt to the current situation of the user, in order to integrate more seamless with the user's environment. For instance, if a mobile phone is used in an area with loud noise like a disco, it should disable sound signalling and only use vibration.
- Contextual resource discovery: using only the own context of a device for services is sometimes adequate, but sometimes more information on the environment is needed. The device should then be able to discover other contextual resources to determine the context of other entities like persons, devices, etc. For instance if the user wants to display a movie and the device has too small a screen, it can look for unoccupied display devices in its neighbourhood. Another example is that one is stuck with a problem, and wants to talk to someone with a similar problem; in this case other context aspects are used than the current location only.
- Contextual augmentation: some services will not only adapt to the current context, but also couple digital information to the environment. Depending on the user's perspective this can be viewed as the digital data augmenting reality or reality augmenting the digital data. An example of digital data augmenting reality is tour-guiding [5] in which the device gives information about the nearby attractions. An example of reality augmenting digital data is when you can view someone's current location - or other context - when you view his/her homepage on the web.

Research groups developed many video processing algorithms for real time dynamic context information extraction and fusion from video sensors [9]. High level information extracted by video, radio, audio and other kind of heterogeneous sensors can be fused together [14] in order to provide the system with a useful representation of the state of the environment and of the subjects interacting within. Dynamic modelling of environment, user localisation, behaviours and events automatic analysis in order to define and collect information of interest are the main tasks of the aforesaid algorithms. Techniques have been developed for counting people in outdoor environments using neural networks [15] or Bayesian

Mobimedia'07, Month 8, 2007, Nafpaktos, Aitolokarnania, Greece.

models from frontal views [7]. Bayesian networks of extended Kalman filters and data fusion algorithms are used in [2] for estimating people waiting on a underground platform. In [11] people are detected in a indoor environment by tracking their heads by using particle filter algorithm.
Many research projects studied methods for recognition of human activities; in [3] multiple, cooperative video sensors are used to provide continuous coverage of people and vehicles in a cluttered environment. Motion features are used directly in [10] rather than try to reconstruct 2D or 3D models of the human body. Proposed method is based on Principle Component Analysis for training and classification. Negative space analysis is used in [17]: propose method is extremely interesting but can be mainly used in a controlled acquisition environment.
Algorithm proposed here is based on tracking of visual features for detecting human and pose estimation in cluttered indoor environments. As in [1] the head's projection onto the image plane is modelled as an ellipse whose position and size are continually updated. The tracking module is automatically initialised by searching for persistent ellipses within the images and linear dynamic Kalman filters are used for predicting heads movements. First and second orders moments are computed in order to estimate postures. The remainder of the paper is organised as follows: section 2 shows overall software architecture of the proposed system allowing multiple camera processing on distributed networks; in sections 3 and 4 automatic methods for extracting contextual data about people and their postures are described. Finally in section 5 results in a real office environment are shown and conclusions are drawn in section 6.

2. SOFTWARE ARCHITECTURE

Figure 1 depicts the general software architecture of the proposed system. Configuration parameters as well as runtime extracted data context are stored onto a remote database powered with a MySQL server. For increasing software modularity, each PC can run a `camera manager` able to process in parallel images acquired from multiple network cameras and to send high level contextual information to a remote `room manager` via TCP socket. A `room manager` is associated to a single room and is able to fuse information extracted from each single camera associated to that room and to store updated contextual data onto the remote database.

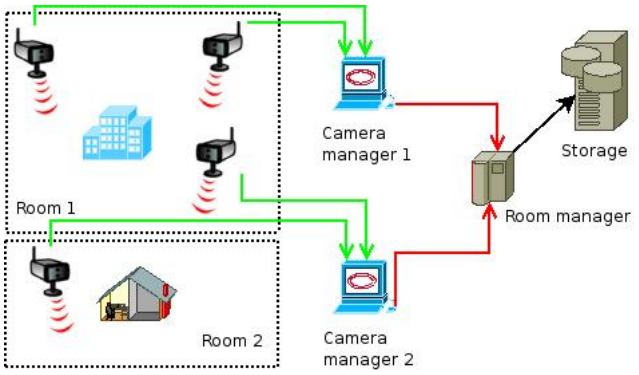

Figure 1: Distributed software architecture schema

2.1 Single camera processing

Each single `camera manager` can be configured in order to acquire and process images from multiple wired or wireless cameras. Different processing threads are dynamically created for single video sources, the number of processed sources per computer depending on processing capabilities of each machine. For each configured source, the system is able to set up an image processing algorithm for extracting information about position of people in the scene and their postures. These information are extracted from video at a very high rate: in fact higher is the number of frames processed per image smaller is the possibility that processing algorithm fail while trying to track detected persons. Information about position and postures of detected humans are sent via TCP socket to a remote room manager that is responsible for fusing and filtering data before storing everything onto a database.

2.2 Multi camera fusion

An application called `room manager` can be configured for receiving contextual data extracted from each `camera manager` that is associated to a particular room. These information are fused and registered to a common reference space related to the monitored room itself. Before writing data to the database, a filtering step is performed for regularising and smoothing extracted data. Third part applications, in fact, will access the database as the basis for delivering ambient intelligence services. Applications supplying context dependent services to the user should exhibit an high level of stability: changes in the status of a room (number of persons, postures, etc.) should happen only if one of the observed variables is changed with a high level of certainty. New contextual data extracted from video are available at a very high rate (at a video frame rate, i.e. between 5 and 25 fps) and the filtering step is needed for avoiding instable behaviours in ambient intelligence services built on the basis of stored data.

2.3 Data storage

Figure 2.3 shows the database that is used for storing configuration and context data information. Data related to the computers involved in the network are stored into `cnf_computer` table. By searching its own ip address into this table each computer is able to find correspondent acquisition and processing parameters into `cnf_camera` and `cnf_videoparams` tables. These tables are connected to `cnf_computer` through foreign keys, such as each `camera manager` application can read directly needed parameters as well as the ip address of the `room manager` that is supposed to receive and fuse context data from that very camera. On the other hand each `room manager` application is able to start and auto-configure itself by looking for its own ip address in the database. A simulation mode was also implemented for testing purposes in order to let each room manager to read contextual data and correspondent timings directly from an `xml` file.
`people` and `people_feature` tables are filled at runtime when new context data are extracted from video. First table contains the number of detected people and the correspondent timestamp for each stored event while second table is used for storing features such as position and posture) for each person in the scene.

3. DETECTION AND TRACKING

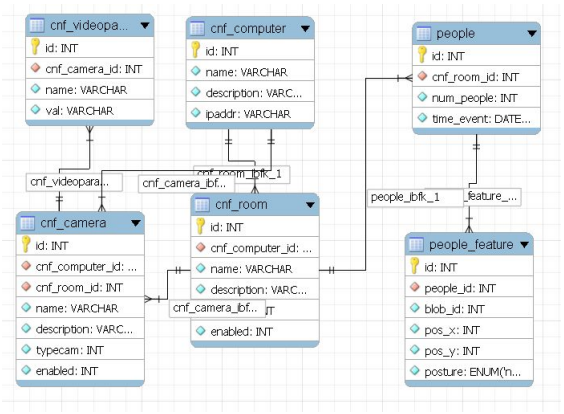

Figure 2: Database diagram

3.1 Ellipse fitting

First step in human detection and tracking is the localization of moving persons as they enter the scene. For this purpose an algorithm for ellipses extraction and fitting has been implemented. Starting from the contours extracted from the image by using the well known Canny algorithm, ellipses are searched through scanning of contour points. By using a contour following algorithm, a set of connected pixels are extracted from the binary Canny image. The least squares method is the most commonly used technique for fitting an ellipse through a set of points. However, while least squares is optimal under Gaussian noise it is very sensitive to severe non-Gaussian outliers, and is therefore unsuitable for many vision applications.

One of the basic tasks in computer vision and pattern recognition is fitting of primitive models to the image data. A technique that is able to fit primitive models to the image data can be used to reduce and simplify complex image data. Ellipses commonly occur in natural scenes, often being formed as the projection of circular objects onto the image plane. They provide a useful representation of parts of the image since they are more convenient to manipulate than the corresponding sequences of straight lines needed to represent the curve, and their detection is reasonably simple and reliable. Conic sections have the form of a second-degree polynomial

$$F(\mathbf{a},\mathbf{x}) = \mathbf{a}^T\mathbf{x} = ax^2 + bxy + cy^2 + dx + ey + f = 0 \quad (1)$$

where $\mathbf{a} = [a,b,c,d,e,f]^T$ and $\mathbf{x} = [x^2, xy, y^2, x, y, 1]^T$. $F(\mathbf{a},\mathbf{x_i})$ is called the "algebraic distance" between point (x_i, y_i) and conic $F(\mathbf{a},\mathbf{x_i}) = 0$. The fitting of a general conic may be approached by minimizing the sum of squared algebraic distances

$$D_A(\mathbf{a}) = \sum_{i=1}^{N} F(\mathbf{x_i})^2 \quad (2)$$

of the curve to the N data points $\mathbf{x_i}$. In order to achieve ellipse-specific fitting polynomial coefficients must be constrained: for ellipse they must satisfy $b^2 - 4ac < 0$. However, this constrained problem is difficult to solve in general as the Kuhn-Tucker conditions [13] do not guarantee a solution. Moreover, the equality constraint $4ac - b^2 = 1$ can be imposed in order to incorporate coefficients scaling into constraint. By introducing this constraints, the fitting problem in 2 can be solved by introducing Lagrange multipliers [6].

3.2 Tracking

By using most persistent ellipses that are supposed to correspond to the heads of people entering the scene, a new tracker is initialized and is associated to moving object. Different tracking approach have been tested and optimized for indoor head tracking. An approach like the one proposed in [1] performs quite well but suffers for variable and potentially low frame rate that is caused by the use of IP wireless video sensors.

The mean shift tracking algorithm [4] is an appearance based tracking method and it employs the mean shift iterations to find the target candidate that is the most similar to a given model that is usually described through a colour histogram. The Kullback-Leibler divergence, Bhattacharyya coefficient and other information-theoretic similarity measures are commonly employed to measure the similarity between the template (or model) region and the current target region. Tracking is accomplished by iteratively finding the local minima of the distance measure functions using the mean shift algorithm.

4. POSTURE CLASSIFICATION

Posture classification is performed by evaluating principal axes of extracted blobs and their orientation. By using an adaptive mixture of Gaussian algorithm as change detection step [16], changed regions are extracted from the scene through a connected components analysis.

Principal axis of extracted regions are computed and compared with the position of extracted heads. This technique allows one to detect the lying posture of the detected person and classify between standard (standing or sitting) and lying persons.

The two second order central moments measure the spread of points around the centre of mass (moments of inertia):

$$\mu_{20} = \sum_x \sum_y (x-\bar{x})^2 f(x,y) \mu_{02} = \sum_x \sum_y (y-\bar{y})^2 f(x,y) \quad (3)$$

However, the pointspread might not be perfectly aligned with the coordinate axes, and thus we get a cross moment of inertia (covariance):

$$\mu_{11} = \sum_x \sum_y (x-\bar{x})(y-\bar{y}) f(x,y) \quad (4)$$

Orientation of the object can be derived from these moments, which means that they are not invariant to rotation. The orientation of an object is commonly defined as the angle relative to the first coordinate axis for which a line through the centroid has the least moment of inertia. This direction can be found by minimizing the moment of inertia around a rotated axis:

$$(\mu|\alpha) = \sum_{\{x,y\} \in R} d^2 = ((x-\bar{x})cos\beta + (y-\bar{y})sin\beta)^2 \quad (5)$$

By deriving this equation and setting to zero, orientation can be shown to be:

$$\alpha = \frac{1}{2} tan^{-1}\left[\frac{2\mu_{11}}{\mu_{20} - \mu_{02}}\right] \quad (6)$$

Figure 3 shows the principal axis (in white) of the blob that corresponds to a fallen person. By looking at figure 4 in can

Figure 3: Principal axis of a lying person

Figure 4: Principal axis in case of a sitting person.

Figure 5: Head detection through ellipse fitting for a single person.

Figure 6: Head detection through ellipse fitting for a group of three persons.

be seen that the orientation of the principal axis of a sitting person can be very similar to the one in the previous figure. In order to reduce the number of false positives an additional control of the direction of the axis (in red in figure 4) going from the position of the head to the centre of mass of the blob is performed.

5. RESULTS

Head detection and tracking algorithms described in section 3 have been tested in a certain number of sequences acquired from a real office. Figures 5 and 6 show results of the head detection step in case of single or multiple persons.

For testing purposes, 20 sequences with 5000 frames each have been considered, containing a total of eight meeting (more than three persons in the room) and seven falling person events. Proposed algorithms performed well in 16 situations: in two sequences one entering person was missed because of low contrast between the head and the background. In those cases extracted ellipses persistences were not sufficient for initializing a new tracker. In other two sequences the system did not classified correctly a falling event because of environmental occlusions: a large part of the body of the lying person was under the desk thus leading to an incorrect estimate of the principal axis of the blob.

To validate the system in a concrete application scenario, we designed a context aware mobile application that uses the information coming from the camera manager and a WiFi camera installed in an office room. The objective was to support the user's communication and safety using the high level context information of the office room obtained trough video analysis.

First we integrated the camera manager and the room manager in a Context-aware platform that we designed [8] to support context acquisition, aggregation and elaboration. In this platform context information is obtained from Context Providers through a Context Broker. The Room Context Provider we developed extracts the information from the storage, updated by the room manager every 10 seconds. As a second step we developed a mobile application that runs on a Symbian S60 mobile phone: the application changes automatically the phone profile to three different possible states - normal, meeting, emergency - based on the information coming from the Room Context Provider about the number of people in the room and their position (standing, sitting or lying on the floor).

The application on the mobile phone communicates with the

Context Broker using a UMTS connection and uses a Bluetooth access point based mechanism to determine the room where the user is located, using information coming from the Location Context Provider available in the platform. Then the application asks periodically to a Web Service what is the target phone profile to be set based on the latest context information. The relation between the room context and the target phone profile is simple: if the Room Context Provider indicates that there are 2 people in the office, the phone automatically switches from 'normal' to 'meeting' profile. The user can configure his desired communication behaviour associated to each profile: for example he can define that in the 'meeting' profile only phone calls from colleagues should be passed, while all other calls should be blocked automatically by the phone and a pre-defined SMS message should be sent to inform the caller that the user is involved in a meeting. Similarly in case the Room Context Provider indicates that only one person is in the room and the person is lying on the floor, the phone profile is set to 'emergency'. This state triggers on the phone an emergency procedure: first an audio message is played by the phone loud speaker, asking to the user to press a key if "everything is ok". In case no input from the user is received, the phone sends automatically an SMS to a pre-defined emergency number to indicate that the user needs help and providing the room where the user is currently located.
The scenario has proved to be effective to validate the tracking algorithms applied in a real case. Also the scenario was useful to fine-tune the update intervals used to provide context information.

6. CONCLUSIONS

This paper described a system that is able to acquire and process images from multiple wireless cameras. Contextual information about number, position and posture of detected persons are extracted from video and used as input for an ambient intelligence application able to deliver context-based services to the users. Proposed result show the validity of the proposed approach.

7. REFERENCES

[1] S. Birchfield. Elliptical head tracking using intensity gradients and color histograms. In *CVPR '98: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, page 232, Washington, DC, USA, 1998. IEEE Computer Society.

[2] R. Bozzano, C. Regazzoni, A. Tesei, and G. Vernazza. Bayesian network for crowding estimation in underground stations. In *Proceedings of the International Conference on Image Processing (ICIP)*, pages 79–82, Bari, Italy, September 1993.

[3] R. Collins, A. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, and O. Hasegawa. A system for video surveillance and monitoring. Technical Report CMU-RI-TR-00-12, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, May 2000.

[4] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(5):564–577, 2003.

[5] N. Davies, K. Cheverst, K. Mitchell, and A. Efrat. Using and determining location in a context-sensitive tour guide. *Computer*, 34(8):35–41, 2001.

[6] A. W. Fitzgibbon, M. Pilu, and R. B. Fisher. Direct least square fitting of ellipses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):476–480, 1999.

[7] S. Harasse, L. Bonnaud, and M. Desvignes. A human model for detecting people in video from low level features. In *Proceedings of the International Conference on Image Processing (ICIP)*, pages 1845–1848, Atlanta, USA, October 2006.

[8] L. Lamorte, C. A. Licciardi, M. Marengo, A. Salmeri, P. Mohr, G. Raffa, L. Roffia, M. Pettinari, and T. S. Cinotti. A platform for enabling context aware telecommunication services. In *Proceedings of the Third Workshop on Context Awareness for Proactive Systems (CAPS 2007)*, Guildford, United Kingdom, June 18-19, 2007.

[9] L. Marchesotti, C. S. Regazzoni, C. Bonamico, and F. Lavagetto. Video processing and understanding tools for augmented multisensor perception and mobile user interaction in smart spaces. *Int. J. Image Graphics*, 5(3):679–698, 2005.

[10] O. Masoud and N. Papanikolopoulos. A method for human action recognition. *Image and Vision Computing*, 21(8):729–743, August 2003.

[11] H. Nait-Charif and S. McKenna. Head tracking and action recognition in a smart meeting room. In *Proceedings of the 4th IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS-ICVS)*, March 2003. CD-ROM proceedings.

[12] J. Pascoe. Adding generic contextual capabilities to wearable computers. In *Proceedings of 2 nd International Symposium on Wearable Computers ISWC*, pages 92–99, 1998.

[13] S. Rao. *Optimization: Theory and Applications*. Wiley Estern, New York, 2nd edition, 1984.

[14] C. S. Regazzoni, R. Singh, and S. Piva. Intelligent fusion of visual, radio and heterogeneous embedded sensors' information within cooperative and distributed smart spaces. In E. Shahbazian, G. Rogova, and P. Valin, editors, *Data Fusion for Situation Monitoring, Incident Detection, Alert and Response Management*, volume 198 of *NATO Science Series*. IOS Press, US, Nieuwe Hemweg 6B, 1013 BG Amsterdam, The Netherlands, March 2006.

[15] C. Sacchi, G. Gera, L. Marcenaro, and C. S. Regazzoni. Advanced image-processing tools for counting people in tourist site-monitoring applications. *Signal Process.*, 81(5):1017–1040, 2001.

[16] C. Stauffer and W. E. L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):747–757, 2000.

[17] M. J. van Vuuren. *Human pose and action recognition using negative space analysis*. PhD thesis, University of Cape Town, August 2004.