GaliTracker: Real-Time Lecturer-Tracking for Lecture Capturing

Elisardo González-Agulla, José L. Alba-Castro
Multimedia Technologies Group
AtlantTIC Research Center, University of Vigo
Vigo, Spain
eli, jalba@gts.uvigo.es

Héctor Canto, Vicente Goyanes
Teltek Video Research
Vigo, Spain
hectorcanto, vgoya@teltek.es

Abstract—This paper describes a fully automated Real-Time Lecturer-Tracking module (RTLT) and the seamless integration into a Matterhorn-based Lecture Capturing System (LCS). The main purpose of the RLT module is obtaining a lecturer’s portrait image for creating an integrated slides+lecturer single-stream ready to distribute and consume in portable devices, where displayed contents must be optimized. The module robustly tracks any number of presenters in real-time using a set of visual cues and delivers frame-rate metadata to plug into a Virtual Cinematographer module. The so-called GaliTracker RLT module allows broadcasting live in conjunction with the LCS, Galicaster\(^1\), or processing offline as a video-production engine inserted into the Matterhorn\(^2\) workflow.

Keywords—Lecture Capturing, Object Tracking, Real-Time

I. INTRODUCTION

In the last few years, lecture recording has become a widely accepted practice in many Universities and institutions around the world [1,2]. Some studies show that recordings composed only by the slides and the lecturer’s speech might be enough for engaging students in a recorded lecture and that the extra cost (production, bandwidth) of capturing the presenter’s image is not worth [3]. Capturing the lecturer image consumes human resources that prevents from a wide spread of these multi-stream recordings [4]. Nevertheless, the video-stream of the lecturer in the recording adds a more natural learning experience by including the non-verbal communication and full emotional expression of the lecturer. So, if the cost of capturing is drastically reduced and the joint visualization with the slides stream is optimized, capturing the presenter image is fully justified. Robustly tracking the speaker and saving/broadcasting the Region Of Interest that contains only the main expressive parts (upper body) makes the system more efficient in terms of bandwidth and visualization. Several attempts have been made in the past to automatically record the lecturer image. Some lecture’s tracking modules rely on accessories to help framing the lecturer, like infrared light emitting microphone [5] or necklace [6], but these solutions are obtrusive, require an IR camera and can not ensure an accurate cropping of the lecturer for applying video-production rules.

In [7] Microsoft Research presented an LCS that used, among other equipment, a pair of coupled wide-angle and PTZ cameras to track the lecturers’ movement. A comparative survey on lectures, recorded both automatically and by professionals, showed a preference of the audience for the professional recordings. The conclusion was that the system was quite useful but some of the videographer rules could not be easily implemented in an automated way. In a posterior work [8] the lecturer tracker was reduced to one PTZ camera that included digital cropping (hybrid speaker tracking). This system has the advantage of reducing one camera but the drawback that several regions have to be manually marked for each room before the tracking starts. Another well-known LCS is the Autoauditorium [9], being its main criticisms that uses several cameras (at least three) for tracking the lecturer and capturing the slides, and the setting up process is cumbersome. The OpenSource project LectureSight [10] includes a Lecturer Tracker module based on moving foreground detection and silhouette analysis to drive a single PTZ camera. This project is very close to ours but differs in several aspects like the tracking technique and the use of the PTZ camera.

In this paper we introduce a lecturer tracking module that works with a cheap HD webcam, that doesn’t need any previous manual initialization, that works in any room where the webcam could be set up to frame the stage area, and that can robustly track and process one or more presenters in real-time. The module delivers metadata in a frame-by-frame basis that are consumed by a virtual cinematographer (VC) in charge of applying videographer rules. The VC can operate on an analog PTZ camera or digitally pan, tilt, zoom and crop the more pleasant and informative image of the lecturer.

\(^1\) Galicaster is an LCS open solution to provide flexible, state-of-the-art system for recording educational multimedia contents, like lectures and conferences. Galicaster project is supported by TELTEK Video Research.

\(^2\) Matterhorn is a free, open-source platform to support the management of educational audio and video content.
The rest of the paper is structured in the following way: Section II pictures the LCS where the tracker is integrated. Section III is the main contribution of this paper, devoted to the technical details of the real-time lecturer tracking module, that we have called GaliTTracker. Section IV explains details on the integration of GaliTTracker in the LCS system, both as a real-time module for broadcasting live and as an off-line module to optimize videographer rules in the processed footage. Finally, section V closes the paper with some conclusions and further research lines.

II. THE MULTI-STREAM AUTOMATED LECTURE CAPTURING SYSTEM AT UNIVERSITY OF VIGO

For the integration of the RTLT we need an multi-stream environment able to integrate an external processing such as Galitracker.

The University of Vigo (UVIGO) with the support of the international institution Campus do Mar [11] have implemented an LCS mainly based on three tools: Matterhorn, Galicaster and Pumukit. These tools are based on state-of-the-art open technologies and are open source – Matterhorn and Pumukit – or open to modify and use – Galicaster. Matterhorn and Galicaster have been developed embracing multi-stream recordings, and Pumukit has been modified to handle dual video streams as well. Fig. 1 shows the block diagram of the LCS.

Matterhorn is a multimedia management platform oriented to educational audiovisual content. We highlight two characteristics of Matterhorn: the adoption of multi-stream content, in the sense we pointed out on Section 1. – camera plus slides plus speech; and the architecture's modularity allowing the integration of new features easily. [12]

Galicaster is an application for the automatized recording of multi-stream lectures. The difference with previous solutions is that it provides an interface for a lecturer or operator to preview and control the recording. It automatically starts and stops the recordings, sending the resulting media to the Matterhorn servers. [13,14]

Pumukit is a media cataloging system developed at UVIGO. It provides a customizable front-end and enhanced capabilities to automatically publish media into YouTube or iTunes-U, among other distribution channels [15]. Inside the LCS, Pumukit is the link between the media production and processing and the target devices for optimized single-stream lecture-slides composition.

During 2012 – two semesters - the multimedia department of the UVIGO implemented a two-staged pilot based on this LCS, totaling 400 hours of published recordings and over 32000 downloads of recorded lectures.

UVIGO's LCS makes the perfect environment for the introduction of the RTLT described in this paper, because of the next characteristics:

• adoption of multi-stream content production,
• distribution to single-stream channels that demand optimization in terms of resolution and size, and
• means to incorporate external applications into its workflow.

III. GALITRACKER: DETECTING AND TRACKING LECTURERS ROBUSTLY

A. System principles

Existing lecturer's tracking systems generally assume several constraints in order to operate seamlessly [5]. A typical constraint is that the lecturer is the only people in the scene (one tracker), that he is moving and gesturing (movement-based segmentation) or that he does not occlude the projection (illumination stability). We propose a tracking module that does not make any hard prior assumption. It is capable to track any number of people in a non-homogeneously illuminated scene regardless their movement or lack of it. This feature is useful in interactive scenarios where more than a lecturer can be in the scene or where the lecturer interacts with a student in the stage area. Rules to decide what lecturer/student is cropped and displayed can be
implemented in a virtual cinematographer module (VC) because all of them are tracked at the same time and their metadata are stored with a unique identifier. The scene is captured by an HD camera (a $100 HD webcam is currently used) with a field of view that covers the stage area where the lecturer walks. Our RTLT module relies on several vision cues working together to obtain a robust lecturer’s tracker and then yields metadata for the VC module. These metadata include the lecturer position, trajectory, head orientation and gesture area. The VC applies videographer rules and extracts the optimal sub-image to append to the VGA captured image (Fig. 2). The detailed operation of the robust tracker will be explained in the next subsections. This way, the only soft assumptions that Galitracker makes to operate are: i) the lecturers stand, walk and talk in the field of view of the wide angle camera, ii) the illumination is enough to see the presenter, and iii) the scene has a static background. Even the third assumption can be eliminated with only a small decrease in the RTLT performance. Figure 2 shows the flow diagram of the RTLT.

Every time a candidate lecturer is detected, a new computer thread is created and a particular detector-tracker tandem is set up. Every thread that is in charge of a lecturer has to keep track of them following the blue sub-flow diagram represented in Fig. 3 and explained in the next subsection. This way, the main program can be devoted to searching new people appearing in the scene at a configurable rate and secondary threads can be devoted to keeping track of previously detected lecturers. A missed track of a lecturer that went out of the field of view is considered a new one when reentering into the scene.

B. Detecting new people

The detection of the lecturer is based on the OpenCV LBP Histogram-based face detector [16] trained both for frontal and profile faces. This algorithm has been optimized to work for this specific scenario, where there cannot be very small faces and static areas cannot contain people, unless one is being tracked. New candidate lecturers in the scene are rapidly evaluated also by using foreground-background segmentation robust to shadows [17].

C. Tracking detected people

The tracking of the lecturers is based on a combination of several visual cues.

1) On one hand, an ad-hoc face tracker is applied: the mean-shift algorithm [18] has been modified to perform more robustly on the presence of illumination changes (room lights on/off, projector light-beam, sun reflection) and to avoid false positives with similar characteristics to the face. This modification was inspired by the Adaptive Background Camshift Tracker, ABCShift [19], in the sense that the tracked object (the face) is modeled as a class conditional color distribution in a chromaticity space (normalized r-g in the current implementation) and the background is modeled as the surrounding area (red and green boxes, respectively, around the face in Fig. 4). This way, the chromaticity model has a discriminative component given by the Bayes’ rule:

\[
P(O / C) = \frac{P(C / O)P(O)}{P(C / O)P(O) + P(C / B)P(B)}
\]

where \( O \) denotes the tracked object, \( B \) the surrounding background and \( C \) the chromaticity of the pixels.

In our case, not only the chromaticity distribution of the background \( P(C/B) \) is updated in every frame, but also the chromaticity distribution of the tracked object \( P(C/O) \). In order to avoid undesirable drifts, the object distribution is only updated when there is a face (frontal or profile) detected.
in the searching area of the tracker. Also, in contrast to the ABCShift approach, our method doesn’t update the size of the tracked object using the distribution information, but only the position (meanshift philosophy). We have concluded that updating the face size using its updated chromaticity distribution is more prone to errors in presence of similar chromaticity neighbor pixels or sudden trajectory changes of the lecturer in the direction of the camera axis. Given that the chromaticity distribution of the face is updated on each positive detection, the size is directly updated to the size of the detected face. The tracker is initialized or updated using a reduced inner region of the detected frontal or profile face (object) and the surrounding area (background). This way, sudden illumination changes over the face (produced, for instance, when the lecturer invades the projector light beam: Fig.4) are less likely to produce track losses.

2) On the other hand, a pattern matching approach has been added to the tracking system to include texture information, which gives robustness to the location of the face regardless of illumination changes. This visual cue weighs the conditional color distribution of the tracked object $P(O/C)$ with a grayscale positively normalized cross-correlation between the current Object area, $O_t$ and the last detected face, $O_{t-k}$:

$$R(O_t, O_{t-k}) = 0.5(\text{normcorr}(O_t, O_{t-k}) + 1) \quad (2)$$

Given that the class-conditional color distribution is statistically independent on the textured grayscale-based cross-correlation, their product can be seen as a joint class-conditional distribution given the chromaticity and the textured pattern of the last detected face. The maximum of $P(O/C)R(O_t, O_{t-k})$ indicates the center of the tracked object.

3) The new position and size of the tracked face is predicted by a Kalman filter [20]. A new observation is then searched in a neighborhood of the predicted location to minimize computational burden in the thread. If a face is detected in that area, the size and texture template are updated and the location is reestimated with the meanshift+template tracker. If a face is not detected, the system relies in the Kalman predictor to control the tracking loop in Fig. 3 for a predefined amount of time. If that time is reached without a new face detection and the gaze is not marked as 180º, the track is considered lost. If the lecturer walks outside the field of view, the thread is also eliminated. The silhouette tracker, explained next, also feedbacks the flag of lost presenter.

4) The last visual cue that Galitracker uses for robust tracking of the presenters is the silhouette. The body (total or partial) of the presenter is detected by means of an adaptive Gaussian mixture-based foreground-background segmentation robust to shadows [17]. This algorithm has the multiple function of reducing false positives of new faces in static areas (yellow loop in Fig.3), tracking a moving body (walking, gesturing) under a tracked face (blue loop in Fig.3), and retaining the presenter’s track when the face is not detected due to self-occlusion or rotation.

Figure 4: Top: ABCShift loosing track. Bottom: our Bayesian approach

D. Feeding the Virtual Cinematographer

In every frame, Galitracker stores metadata that are used by the VC module to apply videographer rules. The metadata that is currently saved is the position, size and gaze direction (-90º, 0º, +90º, +180º) of the face, the center and convex hull of the body, and the estimated trajectory in a 3D view of the stage (position in Z axis is estimated through face size). See Fig. 5 for an example of metadata extraction.

The VC applies a set of predefined rules that can be easily edited and saved for different scenarios (type of rooms, lecturer usual behavior, type of lectures, etc.). The rules can change also depending on the running mode of the LCS. If the lecture is being broadcast live, the VC has to work only with backward metadata, but if the lecture is being saved and broadcasted afterwards, the VC can refine the rules by using also forward metadata. The exact operation of the VC module is out of the scope of this paper.
IV. INTEGRATION OF THE RTLT MODULE IN THE ALC SYSTEM

For the integration of the RTLT module, two main approaches have been proposed: post-processing based, within Matterhorn; and real-time based, within Galicaster. Each approach comes with its pros and cons so it is necessary to evaluate which one is more suitable for each scenario. Now we explain both approaches.

A. Post-processing approach

The integration of the RTLT module within Matterhorn is performed with the assistance of the Execute Service bundle [21]. The Execute Service provides an interface for executing external command-line applications in Matterhorn. It translates the internal Matterhorn data into arguments understandable by the external application; and adapts the output back to the Matterhorn standard format. The Execute Service integrates non-native operations into the load balancing and job orchestrating mechanisms present in Matterhorn.

On the specific case of GaliTracker, the Execute Service provides the camera file and returns a new camera file cropped by the VC module according to the algorithms explained in Section 3.

The main advantage of post processing is twofold: on one hand, the algorithms run in powerful servers, so no restrictions are posed in the complexity of the searching spaces or the precision of the detectors, improving, for instance, the tracking of sudden movements and gestures. On the other hand, as we commented in the previous section, the VC can apply rules taking into account backward and forward metadata in each frame, producing smoother movements in each VC decision. This mode of operation follows a two-pass philosophy, being the first one the tracking of lecturers and the second one the VC.

B. Real-time approach

The real-time approach moves the processing from Matterhorn to Galicaster, therefore adding a couple of restrictions to the system: it is not possible to implement the two-pass mode of operation, and, if we include the RTLT module on the same machine as the capture software, instead of a server, we have to take into account hardware limitations.

The main advantage of real-time processing is that the event can be broadcasted live and displayed readily in mobile devices.

There are a number of implementation variations on this approach. The backward-base VC can pan, tilt, zoom and crop the digital image (current implementation), or send commands to a PTZ camera. In both cases, we could be interested on preserving an unprocessed copy of the lecturer’s video stream for post processing in two-pass mode if needed.

V. CONCLUSIONS

This paper describes a robust and fully automated Real Time Lecturer Tracking module implemented in the Lecture Capturing System designed by Teltek Video Research, Galicaster, extensively used at Vigo’s University. The RTLT module is robust to the changing conditions typically found in classrooms, like several sources of light, different gestures of the lecturer and the apparition of more than one person in the stage. The main purpose of the module is to feed a Virtual Cinematographer module that is in charge of pan, tilt, zoom and cropping the HD image, and build a lecturer image that can be added side-by-side to the slides of the lecture. The RTLT module has been tested in several hours of footage from 4 different lecturers at UVIGO’s repository, and the resulting single-stream lectures have been shown very convenient and comfortable for mobile devices. In any case, an extensive survey on the usefulness of the new displaying format is still missing.

ACKNOWLEDGMENT

This work was partially supported by the Spanish Ministry of Economy and Competitivity under project TEC2012-38939-C03-01, the project CN 2012/260 “Consolidation of Research Units: AtlantTIC”; and the Galician Regional Funds through the Research contract CN2011/019 (Modalidade: Grupos de Referencia Competitiva 2007).

REFERENCES
