Incorporating Geo-Tagged Mobile Videos Into Context-Aware Augmented Reality Applications

Hien To∗, Hyerim Park†, Seon Ho Kim∗, Cyrus Shahabi∗

∗Integrated Media Systems Center, University of Southern California, Los Angeles, USA
†KAIST UVR Lab, Korea
∗{hto,seonkim,shahabi}@usc.edu
†{ilhot}@kaist.ac.kr

Abstract—In recent years, augmented-reality (AR) has been attracting extensive attentions from both the research community and industry as a new form of media, mixing virtual content into the physical world. However, the scarcity of the AR content and the lack of user contexts are major impediments to providing and representing rich and dynamic multimedia content on AR applications. In this study, we propose an approach to search and filter big multimedia data, specifically geo-tagged mobile videos, for context-aware AR applications. The challenge is to automatically search for interesting video segments out of a huge amount of user-generated mobile videos, which is one of the biggest multimedia data, to be efficiently incorporated into AR applications. We model the significance of video segments as AR content adopting camera shooting patterns defined in filming, such as panning, zooming, tracking and arching. Then, several efficient algorithms are proposed to search for such patterns using fine granular geospatial properties of the videos such as camera locations and viewing directions over time. Experiments with real-world geo-tagged video dataset show that the proposed algorithms effectively search for a large collection of user-generated mobile videos to identify top K significant video segments.

Index Terms—Multimedia Content, Geo-tagging, Mobile Video, Augmented Reality

I. INTRODUCTION

In augmented reality (AR), users look at the real-world space through an AR browser where the content is being overlaid on the physical world as objects [11]. Recently, context-aware augmented reality (AR) has been gaining significant attention from both researchers and general users as a new form of media, providing and representing rich and dynamic multimedia content on AR platform. Thus, more and more context-aware AR applications are being developed. However, it is still challenging to create content in current AR browsers. AR browsers such as Layar1, Wikitude2, and Junaio3 are the substitute of Web browsers in the real world, permitting overlay of interactive multimedia content on the physical world or objects they refer to [5]. Although AR is regarded as the next-generation of web browser [12], there is a critical issue for a wide use. There is hardly sufficient content for AR applications. Although recently standardization groups have proposed the ARAF (Augmented Reality Application Format) and the MAR (Mixed and Augmented Reality) reference model [9], [14], most AR content is tightly coupled with certain applications. And, due to the heterogeneous data formats, they are not reusable and interoperable in other AR applications. Therefore, it is necessary to suggest the new solution to provide rich, adaptive, and seamless AR content.

To overcome the scarcity of content, the browsers often adopt multimedia data from traditional social media, i.e., user-generated content such as Twitter, Flickr and Wikipedia [11]. However, the content from web sites does not provide essential spatial metadata for augmentation, which causes imprecise registration to the real world and additional burdensome task through desktop map-based authoring tools. By this problem, the browsers fail to encourage users to use their applications continually [15], [16]. There is a strong need of abundant user-generated content having standardized spatial data for retrieval and visualization on any AR platform.

One important observation about the potential big user-generated content is the availability of mobile videos. As anyone can generate and publish multimedia using their smartphones through social media such as YouTube and Facebook, mobile videos are forming one of the biggest big multimedia data. They become an interesting source of multimedia content for AR applications. Moreover, it becomes practical to collect fine-granular contextual information such as location and orientation of a camera together with mobile video using embedded sensors in a smartphone. Thus, geo-tagged mobile videos have a great potential for various AR applications based on their context.

This study provides a novel way to incorporate geo-tagged mobile videos into AR applications, especially focusing on spatial context. We propose filtering algorithms to effectively select a set of most interesting video segments out of a large video dataset so that the selected scenes can be automatically retrieved and displayed in AR applications. For the filtering, we define an interesting video segment as a sequence of video frames that follow a particular pattern (borrowed from film studies), including tracking, panning, zooming, and arching scenes. In the presence of big video data, it is very challenging to identify a certain pattern, especially in real time. To achieve a highly efficient video filtering and retrieval, we propose algorithms to search for interesting scenes using only geospatial

1www.layar.com
2www.wikitude.com
3https://en.wikipedia.org/wiki/Junaio
properties of videos, i.e., camera locations, viewing directions, and their temporal correlation over time.

We conduct experiments on a real-world dataset with 2397 videos containing 208,978 geo-tagged video frames. Our experimental results show that we are able to identify interesting video segments from a large collection of user-generated mobile videos. Particularly, we found 1789 tracking scenes, 1084 panning scenes, 860 zooming scenes, and 172 arching scenes. The result shows that tracking shot is the most favorable way of capturing mobile videos while the arching shot is quite an uncommon method. Also, panning and zooming are fairly popular. Furthermore, the proposed algorithms are running fast enough to facilitate real-time video retrieval for AR applications.

The remainder of this paper is organized as follows. Section III discusses the preliminaries necessary to present our study. In Section III, we introduce three ways of incorporating user-generated videos and a framework that uses such techniques. Thereafter, in Section IV, we present algorithms for selecting video segments for AR Applications. We present the experimental results in Section V and make the conclusion of the paper in Section VI.

II. PRELIMINARIES

In this section, we first introduce the concept of geo-tagged video and its notations used in this paper. Next, we present an application of adopting crowdsourced user-generated videos with spatial metadata to a context-aware AR browser.

A. Geo-tagged Videos

Mobile videos can be captured at a fine granular level (e.g., frames) and their geospatial metadata (e.g., camera location, viewing direction) are transparently associated with each frame of the video, which is referred as geo-tagged videos. Particularly, we represent a video as a sequence of video frames, or frames for short, and each frame is modeled as a field of view (FOV) [2]. In 2D space, the field-of-view forms a pie-slice-shaped area as illustrated in Figure 1a. We formally define a field of view as follows.

A FOV f is denoted as \((p, d, R, \theta, t)\), where \(p\) is the camera location in \(<\text{latitude,longitude}>\) coordinates, the camera direction \(d\) is obtained based on the orientation angle provided by a digital compass, the camera viewable angle \(\theta\) describes the angular extent of the scene imaged by the camera (\(\theta\) is calculated based on the camera and lens properties for the current zoom level), \(R\) is the maximum visible distance at which a large object within the camera’s field-of-view can be recognized.

The viewable scene of a camera changes as it moves or changes its orientation. In order to keep track of the FOVs of a moving camera over time, we need to record its location \(p\), direction \(d\) and viewable angle \(\theta\) with a certain frequency and produce time-stamped metadata together with time-stamped video streams. Our meta-data streams are analogous to sequences of \((p, d, R, \theta, t)\) quintuples, where \(t\) is the time instant at which FOV information is recorded.

![Field Of View (FOV) model.](image)

Figure 1b depicts FOVs of a 5-seconds video; one FOV is sampled per second.

B. Context-aware Augmented Reality

In an AR browser, virtual content is in general directly registered to each location or object such as points of interest (PoIs). However, a user’s interest may be different from the availability of content, i.e., the density of content may not match to PoIs (Point of Interest). Thus, it is critical to retrieve personalized multimedia from the content based on the context-awareness. Many researchers have focused on the use of context to provide adaptive content to a user [3], [7] using mostly complicated metadata and ontology in a specific place where various sensors and displays are equipped to infer the user’s context and to provide content. Although providing adaptive content service considering user context has been attracting extensive attention in the AR research community, there is little research on methods of filtering and ranking AR content basing on user context. For example, some of them have focused on too simple context data such as GPS or limited user profile to infer user’s context appropriately [20], [4], whereas the other groups have applied complicated methods to infer integrated user’s situation and preference [6], [2]. For a wide and easy use of adaptive AR media service with a large amount of content, it is necessary to build a simple but powerful selection algorithm using the most important user’s context such as location in real world. If the browser does not filter content appropriately, users cannot experience any meaningful information.

The seamless registration of content to a specific real location should be possible in an unprepared environment. Current AR browsers use a couple of registration methods. Marker-based representation is limited to object-based registration (i.e., low utility), while markerless AR requires high computation for tracking physical world (i.e., high computational cost). In addition, for integrating multimedia content to the physical world, users often use conventional authoring systems, and they are not used to georeference content in registering them to right coordinates [8]. The result is that users have great difficulties in generating AR content. Thus, it is essential to build an in-situ mobile application that allows users to create and experience dynamic AR multimedia such as video at the same time.
For developing AR as a new media, it is essential to provide rich content, based on context-aware filtering system. Also, it should integrate virtual content to the real world seamlessly in an unprepared environment. In this respect, we suggest the new way of creating and representing multimedia, specifically user-generated mobile videos, in AR environments, using spatial metadata of crowdsourced videos. It can be a highly cost-effective way to incorporate mobile videos into AR applications without markers and high computational cost.

### III. Incorporating Video Contents with AR Applications

#### A. Three Ways to Utilize Videos

This section describes three ways of using geo-tagged user-generated mobile videos for context-aware AR applications.

1) **Pre-defined Content:** In this approach, points of interest and related video content are pre-defined for a certain AR application. Users (e.g., tourists) can follow a specific path that authors design in advance. This approach is beneficial in exploring places such as historical sites or popular filming sites with specific content for storytelling. For example, as suggested in the previous work [17] about film-induced tourism, people visit the filming locations, where their favorite movies or TV series were shot, and can get more immersive content experience with a carefully planned and designed content. Using spatial metadata of video clips and the order of narratives, the system can guide tourists to the PoIs. As a result, users can understand and enjoy the story even though they have not watched the film or TV series before. However, two limitations of this approach are that the provided content is rigid and hard to be personalized even in the presence of rich user context (i.e., camera location and direction).

2) **On-Demand Retrieval:** This approach enables AR users to freely explore an area while personalized and meaningful content are retrieved and shown to them in an on-demand manner. The on-demand content approach does not require a fixed set of PoIs and registered content; instead utilizing on user contexts such as camera location and camera direction, AR browser searches for top $K$ potentially interesting video segments using the filter-refinement technique. The interesting (or significant) video segments are defined as a sequence of FOVs that follow a particular camera shooting pattern, including tracking, zooming, panning, and arching scenes (Figure 2). These four patterns are widely used in film studies and will be formally defined in Section IV-A. These scenes if detected can be used to show not only interesting but also relevant videos to AR users. For example, if a user is moving, AR browser may prefer to show video segments with a similar movement of camera location; thus, tracking or arching scenes might be more favorable. On the other hand, if a user does not move, panning and zooming scenes may be more relevant. For efficient video retrieval, we will propose algorithms to search for such interesting scenes from a particular video using its geospatial metadata.

<table>
<thead>
<tr>
<th>Position</th>
<th>Direction</th>
<th>Single</th>
<th>Multiple</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single</td>
<td>Zoom</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multiple</td>
<td>Track</td>
<td>Arch</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 2: Four popular camera shots from film studies.

3) **Data-Driven Recommendation:** The third approach recommends the hotspots that cover many video frames to the users. In AR, people are generally interested in popular locations (or hot spots) with more registered content. The indicator of whether a particular location is hot or cold is the number of video frames that are captured at the location. With the data-driven approach, once the hot spots are identified, the system can recommend them to the users when they approach a particular hot spot in the form of a push notification. When a user arrives at the hot spot, the system can query the interesting video segments according to Section III-A2. This approach is useful in guiding AR users to rich-content PoIs, where they would be able to view interesting video segments. Figure 9b shows FOV coverage of the videos collected during two years, 2014 and 2015, at the University of Southern California, showing that several hot spots are automatically recognized. Section IV-C details the algorithm for searching hot spots.

#### B. Visualizing Multimedia Content Through AR

Through context-aware AR applications, users can obtain information interactively and intuitively. Unlike websites where information are provided using virtual web pages, AR...
users consume the information by exploring the real world. For example, as shown in Figure 4a, when a user chooses one video from the recommended list, it represents the virtual video object on the location where it was recorded. With the AR video player, the user not only experiences the real world as the background of the video scene but also interacts with the augmented video, such as moving and rescaling via touch inputs. Additionally, comparing to computer vision techniques for augmenting content, by leveraging only spatial metadata, our approach would not require extensive computation on the mobile device, making it more practicable for a wide use.

In addition, using AR characteristics, the system can provide a novel experience to users. For example, as shown in Figure 4b, the system can guide a user to a PoI or specific direction, using a virtual object such as an arrow. By the AR navigation, users can reduce cognitive effort, comparing to conventional interfaces such as maps or lists. Furthermore, they can have immersive experience related to the scene with additional AR content.

C. A Framework to Search Big Multimedia Data

In order to exploit a big geo-tagged mobile video data in AR applications, we propose a three-step framework as shown in Figure 5 to efficiently search for and identify interesting video segments, which can be presented in an AR application. Using the model described in Section II-A, our approach harnesses spatiotemporal database technologies to support mainly on-demand and data-driven approach in Section III-A2 and Section III-A3, respectively.

First, the framework filters out irrelevant videos based on the user’s current location. A relatively small set of videos in the proximity of the user location is returned from a spatiotemporal range query. Thereafter, the framework refines to identify the most interesting video segments from the returned video set. The significant video segments will be formally defined in Section IV-A based on spatial and temporal relevance. These significant segments are then ranked based on captured time and camera direction. The top K significant segments (K is a small number) are presented to the user via an AR browser. On the other hand, if no data is searched, the framework identifies and recommends nearby hot spots to the user so that he can travel to the hotspot of interests. In either case, the feedback loops are necessary. If the user changes the camera direction, the framework will reselect the significant video segments and update the AR browser, correspondingly. Otherwise, if the user moves to another location, another set of videos are retrieved correspondingly.

IV. SELECTING USER-GENERATED VIDEOS FOR AR APPLICATIONS

This section formally defines interesting scenes (or patterns) and describes algorithms to identify the scenes using spatiotemporal properties of videos.

A. Modeling Significant Video Segments

**Definition 1 (Tracking Scene):** A tracking scene of form \( <t, \theta_{max}> \) is a continuous video segment of length at least \( t \) and the maximum deviation of camera directions during that time is \( \theta_{max} \).

Our semantic hypothesis of defining tracking scene is that if a mobile user points his camera toward a particular direction during a certain period of time, defined by \( t \), it is likely that there is something interesting in the direction of his interest. Figure 6a depicts a tracking scene from geospatial metadata, in which the maximum deviation in camera directions is the angle between \( d_2 \) and \( d_5 \).

**Definition 2 (Zooming Scene):** A zooming scene of form \( <t, \theta_{max}, \delta_r> \) is a special case of tracking scene \( <t, \theta_{max}> \), in which the camera location does not change much. That is, the radius of the minimum bounding circle of camera directions is bounded by \( \delta_r \).

The semantic meaning of zooming scene is that if a user stops for a while and shoots a video toward a specific direction, it is likely that there is an interesting event in front of him. Therefore, we can find significant video segments by searching for the specific pattern, e.g., a period of at least 15 FOVs while the camera movement is minimal. Figure 6b shows an example
of zooming scene, in which all camera locations are enclosed in a small circle with maximum radius $\delta_r$.

The constraint of camera directions on zooming scene can be modified such that the camera directions cover a wide angle over time. We formally define this pattern as panning (or panoramic) scene. Figure 6c represents a model for a panning scene.

**Definition 3 (Panning Scene):** A panning scene of form $<t, \delta_r, \theta_{\text{min}}>$ is a continuous video segment of length at least $t$ while the radius of the minimum bounding circle of camera locations is bounded by $\delta_r$ and the coverage angle is at least $\theta_{\text{min}}$.

Lastly, we define an arching scene in which someone is interested in a particular PoI that is a distance away from the camera locations.

**Definition 4 (Arching Scene):** An arching scene of form $<t, \theta_{\text{max}}>$ is a continuous video segment of length at least $t$ and there exists a non-parallel pairs of camera directions that intersect at the so-called center of the scene and the angle between the center and every camera direction is at most $\theta_{\text{max}}$.

The intersection of the non-parallel pairs of camera directions is a simple yet practical way to define the center of an arching scene. For instance, Figure 6d provides an arching scene, in which its center is the intersection of $d_1$ and $d_2$. This is an arching scene if $\theta_3, \theta_4, \theta_5 \leq \theta_{\text{min}}$. The center needs to be chosen such that its distances to the camera locations are smaller than the maximum visible distance $R$.

### B. Searching and Ranking Video Segments

In this section, we present our algorithms to search for video segments of a particular scene of interest. The algorithms analyze individual video metadata and output a set of video segments that satisfy the definition of the corresponding scene. For example, using the proposed algorithms described below, we find an event as shown in Figure 7b, where someone from a long distance stops and records a group of people singing on a stage. In contrast to tracking and zooming scenes, panning scene likely represents a broad view as shown in Figure 8.

The Tracking algorithm is depicted in Algorithm 1. In Line 8 function $\text{isValidTrackingScene}$ checks if the current segment $s$ satisfy the tracking definition. The function returns $\text{True}$ if the coverage angle of the camera directions is smaller than $\theta_{\text{max}}$. The input of the algorithm is a set of camera directions whose values are between the range $[0,360)$, the minimum duration $t$ and the maximum angle $\theta_{\text{max}}$ is defined in tracking scene. The algorithm sequentially searches for valid candidate segment $s$ (continuous sequence of FOVs of length at least $t$) represented by its start and end indices in $\text{dirs}$. If the algorithm can find a longer valid segment that also satisfy the tracking pattern, we use the longer one instead (Lines 8-9). Otherwise, if the current segment is valid, it is added to the result 11. The algorithm continues to search for the next candidate segment by moving its start index $s$.

Particularly, Lines 14-17 finds the largest index $j$ of the current segment $s$ that invalidates the tracking scene.

### Algorithm 1 TRACKING ALGORITHM

1: Input: camera directions $\text{dirs}$, min duration $t$, max angle $\theta_{\text{max}}$
2: Output: a set of tracking scenes: $\text{segments}$
3: Init $\text{segments} = []$, $i = 0$
4: Init $s = \{\text{start} = 0, \text{end} = 0\}$ \{s is current candidate segment with start and end indices\}
5: While $i < \text{len(dirs)}$:
6: \hspace{0.5cm} If $s.\text{end} - s.\text{start} < t - 1$ and $s.\text{start} < \text{len(dirs)} - t + 1$
7: \hspace{1cm} $i = s.\text{start} + t - 1$ \{jump i if possible\}
8: \hspace{0.5cm} If $\text{isValidTrackingScene}(\text{dirs}, \{s.\text{start}, i\}, \theta_{\text{max}})$:
9: \hspace{1cm} Update $s.\text{end} = i$ \{extend s to include i\}
10: \hspace{0.5cm} $\text{FOV}$
11: \hspace{0.5cm} Else:
12: \hspace{1cm} If $s.\text{end} - s.\text{start} \geq t - 1$: \{found a scene\}
13: \hspace{1cm} $\text{segments}.\text{append}(s)$
14: \hspace{1cm} $s.\text{start} = s.\text{start} + 1$
15: \hspace{1cm} For $j$ from $i - 1$ to $s.\text{start} - 1$
16: \hspace{1cm} If not $\text{isValidTrackingScene}(\text{dirs}, s, \theta_{\text{max}})$:
17: \hspace{1.5cm} $s.\text{start} = j - 1$
18: \hspace{1.5cm} break \{found an invalid scene\}
19: $i = i + 1$

The pseudo codes of the other algorithms are similar to the Tracking algorithm, except checking function in Lines 8 and 15 and the input in Line 1 of Algorithm 1 need to be modified correspondingly. For example, another parameter to be considered in the Arching algorithm is the maximum angle $\theta_{\text{max}}$. With the Zooming algorithm, we need to consider not only the maximum angle $\theta_{\text{max}}$ but also the maximum radius $R_{\text{max}}$ of the minimum bounding circle of the camera locations. When compared to the Zooming algorithm, the Panning algorithm has the minimum angle $\theta_{\text{min}}$ as input instead of the maximum angle.

### C. Detecting Hotspots

In this section, we present an algorithm for finding fine granular hotspots from a large collection of video metadata. A hotspot is a small region (i.e., 100 square meters) that is covered by more than $\theta_{\text{hot}}$ FOVs. A cell is covered by a FOV if the cell’s center is within the FOV area. For example, in
hotspots, popular places where many people visited frequently (e.g., Tommy Trojan at USC), and specific events happened at the campus and many videos were collected during a short time period (i.e., two-days event of Los Angeles book festival in 2014).

Algorithm 2 Hotspot Algorithm

1. Input: FOVs $F = \{f_1, f_2, \ldots\}$, minimum cell size $\text{minS}$
2. Output: number of covered FOVs per quadtree partition
3. Initialize root cell $\text{ROOT.data} = F$, $\text{ROOT.count} = |F|$
4. Initialize queue of candidate cells to be split $Q = \{\text{ROOT}\}$
5. While $\text{size}(Q) > 0$
6. Remove a cell to be split $C \leftarrow Q$
7. If $C$ is covered by multiple FOVs and its size $> \text{minS}$:
8. Split $C$ into four equal quadrants $\text{NW, NE, SW, SE}$
9. Move data from $C$ to its children
10. Compute the number of covered FOVs for each child, $\text{count}$
11. Update queue $Q \leftarrow Q + \{\text{NW, NE, SW, SE}\}$

V. PERFORMANCE EVALUATION

We conducted several experiments on a real-world dataset to evaluate the performance of our proposed approaches. Below, we first discuss our experimental setup and then we present our experimental results.

A. Experimental Methodology

We used the real-world geo-tagged mobile video dataset that has been collected as part of the MediaQ (mediaq.usc.edu) and GeoVid (geovid.org) projects [13], both developed by the authors. The dataset has been collected by volunteer users and its statistics are summarized as follows: 2,397 videos containing 208,978 video frames, that were geo-tagged (i.e., both GPS locations and compass directions at fine-grained intervals), collected by 289 users (e.g., students, faculties) over a period of 10 years (2007–2016). Table I shows the overall statistics of the dataset. Most of the videos are recorded by users in a casual walking mode. The camera moving speed is 4.5 km/h on average, and the camera rotation speed is 10 degrees/sec (i.e., the azimuth angle $\theta$ changing speed). The average FOV sampling rate is 1.03 FOVs per second, and each video is associated with 74.16 FOVs on average.

In all experiments, we varied the maximum angle $\theta_{max} \in \{5, 10, 15, 20, 25, 30, 35, 40, 45\}$ degrees, the minimum angle $\theta_{min} \in \{60, 90, 120, 150, 180, 210, 240, 270, 300\}$ degrees and the minimum duration $t \in \{5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60\}$ seconds. Default values are shown in boldface. In addition, we fixed the minimum cell size to 10 meters. The experiments were run on an Intel(R) Core(TM)i5 CPU at 2.30GHz with 4GB of RAM.

B. Experimental Results

We first report our results for selecting significant video segments. Then, we show the results for detecting hotspots.
### TABLE I: Overview of the dataset.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of videos with geo-metadata</td>
<td>2,397</td>
</tr>
<tr>
<td>Average length per video with content (sec)</td>
<td>72.14</td>
</tr>
<tr>
<td>Average camera moving speed (km/h)</td>
<td>4.5</td>
</tr>
<tr>
<td>Average camera rotation speed (degrees/sec)</td>
<td>10</td>
</tr>
<tr>
<td>Total number of users</td>
<td>289</td>
</tr>
<tr>
<td>Average number of videos by each user</td>
<td>8.29</td>
</tr>
<tr>
<td>Total number of FOVs</td>
<td>208,978</td>
</tr>
<tr>
<td>Average number of FOV per second</td>
<td>1.03</td>
</tr>
<tr>
<td>Average number FOV per video</td>
<td>74.16</td>
</tr>
</tbody>
</table>

1) **Selecting Interesting Video Segments:** Figure 10a compares the number of scenes found for each scene type. Particularly, for the same set of specific parameters ($t = 15, θ_{max} = 15, θ_{min} = 120$), we identified 1789 tracking, 1084 panning, 860 zooming, and 172 arching scenes. This result shows that tracking is the most popular way of capturing mobile videos while arching is quite uncommon. Also, panning and zooming are fairly popular. This result also shows that mobile users favor single-direction videos (i.e., tracking and zooming scenes) over multi-directional videos (i.e., panning and arching scenes). On the other hand, both single-position and multiple-position videos are widely captured. Furthermore, Figure 10b compares the average search time per each proposed algorithm. Interestingly, the search time of the most popular Track scene is the shortest while that of the least popular Arch scene is the longest. The overall search time is short, which demonstrates the practicality of the proposed algorithms.

![Fig. 10: Algorithm performance ($t = 15, θ_{max} = 15, θ_{min} = 120$).](image)

(a) The number of scenes found  
(b) Searching time

Next, for each scene type, we count the number of scenes found by varying one parameter ($t$ or $θ_{max}$) while fixing another ($θ_{max}$ or $t$), respectively. Figure 11a illustrates the results by varying the minimum duration $t$. We observe that with the increase of $t$, much less tracking scenes are found, which is intuitive. The reason is that the higher the minimum duration becomes, the more restrictive a tracking scene is. In the same fashion, Figure 11a shows the results while varying the maximum angle $θ_{max}$. As expected, a higher $θ_{max}$ yields detection of more tracking scenes as the constraint on single direction becomes less restrictive.

![Fig. 11: Number of tracking scenes found when varying $t$ and $θ_{max}$.](image)

(a) Vary minimum duration $t$  
(b) Vary maximum angle $θ_{max}$

We measure the number of arching scenes found while varying $t$ and $θ_{max}$ (Figure 12). The figure shows that the number of detected scenes decreases sharply as we increase $t$ (12a) while it increases gradually when $θ_{max}$ increases. Similar results are observed for the zooming scene in Figure 13.

![Fig. 12: Number of arching scenes found when varying $t$ and $θ_{max}$.](image)

(a) Vary minimum duration $t$  
(b) Vary maximum angle $θ_{max}$

Figure 14 presents the results for the panning scene while varying $t$ and $θ_{min}$. Similar to the previous results, the number of the detected panning scenes decreases greatly with the increase of $t$; however, increasing $θ_{min}$ does not significantly lower the number of scenes found. This unexpected result indicates that most of the panning scenes are near 360-degree view. The reason is that our dataset [13] contains many videos with horizontal 360-degree view as a part of data collection for our study in generating panoramic images from geo-tagged videos [10].

2) **Detecting Hotspots:** The construction time of Algorithm 2 is 3.4 seconds with the dataset, which shows the practicality of the hotspot algorithm. Figure 15a shows the number of hotspots detected while varying the number of minimum FOVs, $δ_{hot}$. Clearly, the number of detected hotspots increases as $δ_{hot}$ decreases. Particularly, there are 60 hotspots with more than 60 FOVs (each FOV has roughly one second of video content). Most of these locations are in Los Angeles, Munich and Singapore where the authors collected videos intensively. Furthermore, we fix $δ_{hot}$ to 10 and report the distributions of hotspots with respect to the ranges of FOV counts in Figure 15b. The graph seems to follow the power-law distribution. Particularly, we observe that 10% of hotspots with more than 60 FOVs contains 26% of the total 24,006 FOVs.

![Fig. 15: Number of hotspots detected while varying $t$ and $θ_{min}$.](image)

(a) Vary minimum duration $t$  
(b) Vary maximum angle $θ_{max}$

![Fig. 15: Distributions of hotspots with respect to the ranges of FOV counts.](image)

(a) Vary minimum duration $t$  
(b) Vary maximum angle $θ_{max}$

**VI. Conclusion**

This paper investigated three general approaches for incorporating video content into AR applications; pre-defined content, on-demand content, and suggested content by hotspots. To efficiently incorporate mobile videos into AR applications,
we proposed algorithms to identify both interesting video segments and hotspots. Our experimental results with a large real-world dataset demonstrated that the proposed algorithms are fast and able to find interesting video segments. Also, the hotspot algorithm efficiently found all hotspots in the dataset. As future work, we will extend this study to consider user mobility and ranking of video segments. The intuition is that when a user moves, AR browser might favor certain patterns such as tracking and arching scenes while the other scenes can be more favorable when the user is not moving.

VII. ACKNOWLEDGEMENT

This research has been funded by NSF grants IIS-1320149, CNS-1461963 and the USC Integrated Media Systems Center. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of any of the sponsors such as NSF.

REFERENCES