ACTIVE FEEDBACK FOR ENHANCING THE CONSTRUCTION OF PANORAMIC LIVE MOBILE VIDEO STREAMS

Motaz El-Saban, Mahmoud Refaat and Ayman Kaheel

Cairo Microsoft Innovation Lab

ABSTRACT

Constructing a panoramic video out of multiple incoming live mobile video streams is a challenging problem. This problem involves multiple users live streaming the same scene from different angles, using their mobile phones, with the objective of constructing a panoramic video of the scene. The main challenge in this problem is the lack of coordination between the streaming users, resulting in too much, too little, or no overlap between incoming streams. To add to the challenge, the streaming users are generally free to move, which means that the amounts of overlap between the different streams are dynamically changing. In this paper, we propose a method for automatically coordinating between the streaming users, such that the quality of the resulting panoramic video is enhanced. The method works by analyzing the incoming video streams, and automatically providing active feedback to the streaming users. We investigate different methods for generating the active feedback and presenting it to the streaming users resulting in an improved panoramic video output compared to the case where no feedback is utilized.

Index Terms— Panoramic Video Creation, Active Feedback, Real-time, Mobile Video Capturing, User Study.

1. INTRODUCTION

Today, mobile phones are more ubiquitous than ever. A large percentage of these mobile phones have video capturing, network connection and different sensors capabilities. The rapid increase in the number of mobile phones and their capabilities has led to the emergence of many new scenarios and applications. Some of these applications already made it to the consumer level, like the live mobile video streaming services [1, 2]. These online services allow users to capture and stream live video to a website where other users/friends can watch the video at the same time it is being captured. Recently, research work has been carried to build services on top of the mobile live video streaming such as stitching in real-time incoming streams as in the case of [3]. The goal is to produce a panoramic video stream out of the multiple video streams captured at the same location. The promise of such a stitching service is hindered by the fact that users, without coordination among themselves, can produce streams that are either unstitchable or stitchable with large amount of overlap between the captured video streams; hence the benefit out of combining multiple streams is diminished. The underlying assumption here is that every user is capturing the scene with little information about other users viewing volumes. In this paper, we introduce the concept of Active Feedback which utilizes network connection capabilities of mobile phones for providing hinting information for the capturing users. The main objective of the feedback information is to maximize the probability of a successful panoramic video result. The intention here is to provide a per-user feedback to guide that particular user. Towards that end, the system receives the incoming user stream along with other streams, analyzes them, generates the feedback, and send the feedback to the streaming user in real-time to help him improve the live generated video. It is worth noting here that the real-time constraint on analyzing the incoming streams is a mandate for the feedback to be timely. To the best of our knowledge, the proposed approach for improving panoramic video quality through user interaction has not been attempted before in the literature. The key technical contributions in this paper are:

- constructing a real-time feedback system for enhancing panoramic video construction quality[1].
- investigating different methods for triggering, computing and presenting the feedback
- conducting a user study to evaluate different feedback aspects and show the gains achieved by utilizing interaction

The rest of this paper is organized as follows. Section 2 reviews related work. Section 3 presents an overview of the proposed active feedback stitching system. Section 4 describes details of the proposed implementation of active feedback. Section 5 describes the experimental setup, the datasets used for evaluation and the results. Finally, section 6 draws some conclusions and proposes directions for future work.

2. RELATED WORK

We review related work relevant to this paper along two main areas: a) video stitching and b) providing feedback for users while shooting videos. In the area of video stitching, a number of research publications have addressed the problem of creating a panoramic image out of a single video [4, 5, 6]. The main idea is to stitch together video frames from a single video feed to generate one wide panoramic image. In this paper, we deal with creating a panoramic video out of multiple videos such as in [7, 8]. In most of the previous work on video stitching, the techniques are based on image stitching which is a very well researched area [9, 10]. Howver, there are main difference between image stitching and video stitching as the latter has unique features that can be used for the stitching process, such as audio and the temporal information. Besides, video faces more challenges like moving cameras, lack of consistency in terms of stitching individual frames or dropping some of them.

---

[1] A sample video screen capture of the running system is included in the supplementary material with this paper for the reader to appreciate system speed.

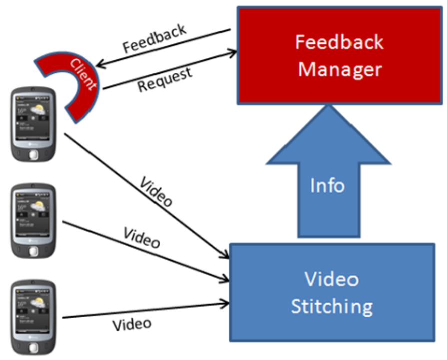

Fig. 1. System Architecture with contributions made in this paper colored in red

The second area of related work is on providing feedback to users while they are shooting a video, such as the interesting work in [11, 6]. However, [11, 6] focused on generating a panoramic image using videos. The work here focuses on generating panoramic video using videos. Another interesting work is presented in [12] where the authors generate a wide video texture output from a single panning video with minimal user interaction. Though, there are some similarities with this work, the end goal is the not the same and the work in [12] does not consider any user feedback while generating the actual stitched output. An alternative form of feedback commonly available nowadays is in digital cameras offering a panoramic picture mode. These devices help the shooter in capturing successive pictures to produce a winder scene by giving suggestions on where to snap the next picture. Nevertheless, the type of feedback provided in these digital cameras is still quite primitive compared to the ones explored in the presented work and there is no published work evaluating the generated feedback.

3. SYSTEM OVERVIEW

In Figure 1, the architecture of an end-to-end generic video mobile video streaming system is shown augmented with real-time stitching of incoming video streams and active feedback information to streamers. The server receives all streamed videos, stitches them together and provides the output panoramic video. We briefly describe the utilized video stitching method as the focus of this work is not on how to generate the stitched video output, rather on how to utilize user feedback to generate a better quality stitching. This concept can be applied to any stitching method such as the excellent works in [6, 4, 10]. In this paper, we propose stitching timely synchronized videos using a frame-by-frame basis as in [3][2]. For each time synchronized pair of frames, the stitching algorithm implements the two main steps, alignment and compositing. The purpose of alignment is to estimate a geometric transformation matrix relating the frame pair. It involves extracting interest point features from both images, and matching them together. In our implementation, interest points are corners extracted by Shi-Tomasi rotation invariant corner detector [13]. For the sake of achieving a real-time performance without much sacrifice to effectiveness we chose Shi-Tomasi's detector with a feathering scheme for composition [14], though are more elaborate methods for interest point extraction and composition such as [15, 16].

On top of this generic video streaming and stitching architecture, we are proposing the use of a feedback channel that can improve the resulting panoramic video. In Figure 1, the proposed contributions are colored in red and are composed of two main parts, the mobile client and the feedback manager.

Mobile client A new mobile component is added to the mobile client and has two main responsibilities: a) pool the server frequently, asking for feedback, and b) retrieve the feedback signal from the server, and present it to the user. We have investigated different rates for feedback generation as well as different presentation schemes.

Feedback Manager The feedback manager component is the main part responsible for generating the feedback. It receives video streams and other information from the video stitching component, and generates suitable feedback for each user.

4. ACTIVE FEEDBACK

Active feedback is the concept of providing in real-time feedback information to the video shooter in order to improve the quality of the final stitched video. We investigate the concept of active feedback for improved video panorama along a number of dimensions: a) goals for providing feedback, b) different scenarios for video shooting, c) feedback triggers, d) feedback implementation and e) feedback presentation. For simplicity, we first consider the base case when only two users are using the system. Scaling to more than two users is addressed next.

4.1. Goals

The goal of Active Feedback is to provide a better viewing experience to the end user watching the stitched video. More specifically, we define a better viewing experience as either an increase in the width of the stitched videos by minimizing the amount of overlap in case of stitchable pairs) or a guidance given to users to render the videos stitchable in case of non-stitchable videos. In the latter case, feedback would be useful if the streams used to be stitchable at some previous time point.

4.2. Video shooting Scenarios

The most important video shooting cases where feedback could be applied are:

- Two users are shooting two views of the same scene that have an amount of overlap and the two views are stitchable. We call this case ”stitchable”.
- Two users are shooting two views of the same scene that had an amount of overlap between them and the two videos used to be stitchable. Then one of the users moved his mobile away from the overlapped area and the videos became non stitchable. We call this case ”used to be stitchable”.
- Two users are shooting two views of the same scene with no overlap between them. We call this case ”never stitched before”.

---

[2] Temporal information can be easily integrated to avoid frame-by-frame stitching as proposed in [7]

We note that in the above cases, feedback is not necessarily generated; rather these are plausible situations for feedback. Generating feedback will depend on triggers discussed next.

4.3. Triggers

There are many possible events that could trigger the feedback manager to create and send feedback to clients. In our investigation, we have experimented with a number of trigger variants with each one experimentally evaluated on a collected real-dataset.

OverlapRatio: The amount of overlap between two videos exceeds 30% of one of the videos. The percentage 30% was experimentally validated to be the minimum required overlap to perform successful video stitching [3].

IPLocation: One, or more, of the interest point used in stitching previous frames is about to get out of the overlapped area.

MotionTracking: The videos became not stitchable because one of the users has moved his camera away.

Initial Condition: If there is no stitching happening and we know that users are located within proximity.

In our experiments, we will have two set of triggers evaluated. The first set is a combination of triggers 1, 3 and 4 referenced as *OverlapRatioSet*. The other set is a combination of triggers 2, 3 and 4 and referenced as *IPLocationSet*.

4.4. Implementation

The instantiation of the Active Feedback concept involves two main aspects. The first aspect involves how the feedback is being generated. The second describes how the feedback is being delivered from the server to the mobile clients. In all feedback cases, the feedback manager generates feedback signals and instructions at the server side while the mobile clients pull the server on a regular basis for feedback. We use a pull mode for communication instead of a push mode as this mode allows the clients to have more control over the feedback rate according to their own capabilities [17].

4.4.1. Active Feedback triggers

OverlapRatio: Using the transformation matrix between two videos, we calculate the amount of overlap between both videos. If the amount of overlap is less than or equal 30%, we send instructions to the video shooters to increase the overlap. We perform that by requesting the user on the right to move the camera to the left and the user on the left to move the camera to the right. It is worth noting that we could generate feedback to one user only, but for simplicity we issued it for both users leaving selection of which user to send feedback to for future work. In the implementation, the clients use the HTTP protocol for retrieving the feedback. We have implemented an ASP.NET HTTP module as a disk-based approach wouldn't be able to handle the reader/writer synchronization problem between the feedback manager writing the feedback, and the mobile clients reading it.

IPLocation: We keep track of matched interest points in the area of overlap and caused the generation of the transformation matrix between the two videos. We track the motion of these interest points using Luca Kanade optical flow [18] and verify if any of them moved out of the overlap area due to camera movement. In such a case, we send feedback to the users asking them to move in a direction opposite to their last motion. It is worth noting that there has been a significant body of work on optical flow methods with [19] and [20] providing good survey and comparative evaluation of existing methods. For our purpose, Luca Kanade provided sufficiently acceptable results.

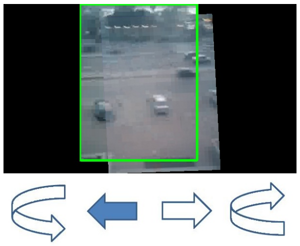

Fig. 2. Different feedback presentation methods. In this particular example, both forms of feedbacks are shown and the arrows suggest to the user to move to the left as there is chance to increase the output frame width.

MotionTracking: we keep track of camera motion using also Luca Kanade optical flow. If the videos cease to be stitchable, we retrieve the positions of both cameras right before stitching was unsuccessful and give feedback to users requesting them to return to the last known stitchable position.

InitialCondition: Initially, if the two videos are not stitchable, we send back a composite image of the two streamed frames and put them side-by-side as a feedback for the users. This gives the users a chance to know where other users are located within proximity.

4.4.2. Feedback presentation

An important aspect of the active feedback concept is the way the feedback will be presented to the end user on his device. There are a number of alternatives for presentation including: a) suggested movement direction arrows, b) a feedback image and c) a combination of both. As shown in Figure 2, a set of arrows advise the user to move left, right, rotate clockwise or anti-clockwise[3]. Use of a feedback image is also illustrated in Figure 2 where a snapshot of the user's video and the other user's video is shown. In this particular case, the two videos can not be stitched together.

4.5. SCALABILITY

To extend Active Feedback to more than two videos, we estimate a global alignment to understand the initial position of all videos. The global alignment determines the order of the videos (right to left in our case as we assume a $1D$ camera arrangement). We then run the feedback generation algorithm on the mobiles in pairs and propagate the feedback to other mobiles. For example, consider L mobiles with order $n_1, n_2, \ldots, n_L$. n_1 is being the left most mobile and n_L is being the right most mobile. We apply the Active Feedback techniques on the pair (n_1, n_2). If the feedback manager decides that n_2 should

---

[3] we assume that the captured scene is at a large enough distance such that in-plane camera motion would be sufficient. If the assumption is violated, motion parallax problems will arise. Dealing with these issues are left for future work.

move a distance $t_{x_{21}}$ (this is the distance in x axis that mobile phone number 2 should move with respect to mobile 1). Then we apply the Active Feedback techniques on the pair (n_2, n_3). If the Feedback Manager decides that n_3 should move a distance $t_{x_{32}}$, then the total amount of movement for n_3 will be $(t_{x_{21}} + t_{x_{32}})$. Note that t_x could be positive or negative. Clearly, the computational complexity of the feedback generation process is much lower compared to the video stitching pipeline itself. Hence, the proposed scalability algorithm could scale up to the maximum number of streams supported by the stitching algorithm which presents the computational bottleneck in this case.

5. EXPERIMENTAL RESULTS

The reported experiments aimed at answering three main question categories[4]. The first category is related to the amount of improvement gained by introducing the Active Feedback concept to the process of live mobile video stitching. More specifically, we want to answer the questions: a) does stitching recall and precision increase after using feedback, b) is the video generated wider, c) which set of triggers is preferred and d) how does feedback affect stitching consistency.

The second category of questions is related to the way the Active Feedback is presented to the user, specifically how useful are the arrows and the feedback images with bounding boxes. The last category is related to the evaluation of the quality of the feedback itself. Specifically, we want to investigate a suitable rate for the feedback generation and how useful is the feedback. In other words, should the user follow it or not.

5.1. Datasets

For the purpose of evaluating the stitching output, a number of authors have proposed to use synthetically distorted images with known transformation matrices as in [21]. The obvious drawback of evaluation using synthetic datasets is that they do not model in full the real case scenario when videos are captured. For that reason and due to the lack of publicaly available datasets for evaluating the proposed active feedback consept, there is a need for collecting our own dataset. In the collected set, special care has been taken to cover different capturing conditions such as day and night and textured and structured scenes. Table 1 gives a description of each video set and its total number of frames. Figure 3 shows sample images from this dataset.

For each scene, two users were standing in front of the view, holding their mobile phones, and shooting the videos. The users were instructed to focus on the same scene (example the small green park in the first video see Figure 4.a). Nevertheless, we gave them the freedom to horizontally and/or rotate the mobile phones (in-plane only) as they deem suitable. For the each scene, we ran the experiment three times, once without using Active Feedback, and twice with Active Feedback but with two different implementations (*OverlapRatioSet* vs. *IPLocationSet*). For the runs that contain the feedback, we instructed the users to try to follow the feedback as much as possible. We saved the resulting video files and carried a number of analyses as detailed next.

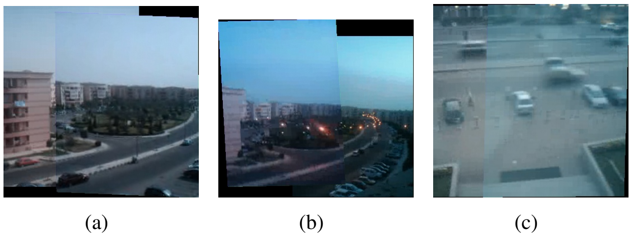
(a) (b) (c)

Fig. 3. Sample stitched results from the dataset.

5.2. Performance gains by Active Feedback

The goal of these set of experiments is to evaluate the different set of proposed triggers for active feedback. We compared *OverlapRatioSet* to *IPLocationSet* schemes using the measures: a) precision, recal and $F1$ measures of the stitched frames, b) output video size (and overlap area) and d) video stitching consistency.

5.2.1. *Stitching precision and recall*

We estimated the stitching precision, recall and $F1$ measures on the data set. The precision (Pr) is calculated as the ratio of the number of correctly stitched pairs by the algorithm to the total number of claimed-to-be stitched, while the recall is calculated as the ratio of correctlt stitched pairs to the total number of frames that can be stitched by a human judge. Finally, $F1$ measure summarizes performance into a single metric as $\frac{2PR}{P+R}$.

5.2.2. *Video output size and overlap area*

We have calculated the average width of output videos in both cases where feedback is utilized and not utilized. While the width of a single video stream is 240 pixels, the average of the stitched video streams is 339 pixels using Active Feedback with *OverlapRatioSet*, 301 pixels using active feedback with *OverlapRatioSet* and 295 pixels without using Active Feedback. This means that Active Feedback increases the width of the final video by 15%. As width in terms of pixels is an absolute measure, we sought a relative measure to normalize for the initial video size. We have experimented with the normalized average area of overlap (percentage wise) in both cases. It was calculated as follows:

$$O = \frac{O_p}{w_1 * h_1 * w_2 * h_2} * 100\% \quad (1)$$

where O is the overlap percentage, O_p is the number of pixels that are in the overlap area, w_1 and h_1 and w_2 and h_2 are the width and the height of the first and second videos respectively. The smaller the percentage overlap the larger is the resulting output video. Using Active Feedback the overlap ratio was found to be 58.8% and 72.8% with *OverlapRatioSet* and *IPLocationSet* respectively, while with no active feedback, it was found to be 73.5%. This indicates a 20% decrease in overlap in the the output video.

5.2.3. *Video Stitching Consistency*

Another method for evaluating video output quality is to measure how consistent video stitching is generated. To elaborate more, Fig. 4 shows 2 graphs. The horizontal axis is time while the vertical

---

[4] We provide in the supplementary material with this submission the set of frames that were used in the human evaluation study to aid in understanding what the human judges were asked to evaluate.

Table 1. Dataset description

Title	Description	Number of frames
LoungeArea	Indoor, close scene, light	396
ParkAtCity1	Daylight, distant scene, slow motion	568
ParkAtCity2	In shade, distant scene	592
NileRoad1	Daylight, slow motion, distant scene	357
NileRoad2	In shade, moving cars and people, distance scene	647

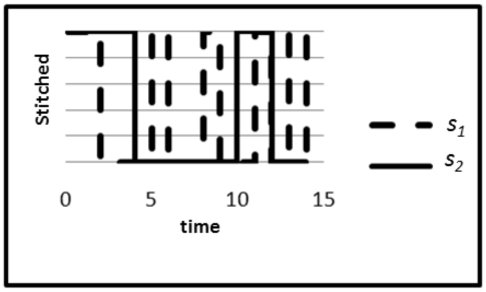

Fig. 4. Stitching consistency example. The vertical axis is a binary measure of whether stitching was produced or not for a given time point.

one represents whether the frames are stitchable or not, with value 1 meaning frames were stitchable at time t while value 0 means frames were not stitchable. In theory, both graphs have the same percentage of the stitched frames. However, s_2 is more convenient, because the switching frequency between stitching and non-stitching states is less than s_1 and hence less annoying or confusing to a watching user. Hence, it is important to measure if Active Feedback decreases video stitching consistency. We propose to use a total variation (TV) like measure for a video V having N frames to capture the notion of consistency in a video using:

$$TV(V) = \frac{\sum_{t=1}^{N} |s(t) - s(t-1)|}{N} * 100\% \tag{2}$$

where $s(t)$ equals 1 it frames at time t were stitchable, and 0 otherwise and N is the total number of the frames.

The results in Table 2 summarize the performance of the different suggested active feedback methods against the case where there is no feedback. A clear difference is visible across all metrics with the *OverlapRatioSet* being the best according to the amount of overlap in the output frames and a very close second in $F1$ measure. The improvement in $F1$ is almost 20% compared to the baseline of no feedback. It is worth pointing out that a possible explanation to why *IPLocationSet* is better than *OverlapRatioSet* in terms of recall is that *IPLocationSet* tries to keep all interest points found in the overlapped area. If there is an interest point on the leftmost part of both videos, it will always try to include this point. This will definitely increase the number of stitched frames. As for the stitched output consistency, using Active Feedback, the average TV is found to be 17% and 14% using *OverlapRatioSet* and *IPLocationSet* respectively, as compared to 15% when Active Feedback is not used. These results indicate that Active Feedback doesn't hurt consistency of the stitched videos. Table 2 summarizes all results using the metrics discussed in this section.

Table 2. Stitching results summary comparing different active feedback methods to the case where there is no feedback.

	Pr	Re	F_1	Width	Overlap
NoFeedback	0.95	0.49	0.65	295	73.48
OverlapRatioSet	**0.97**	0.65	0.78	**338**	**58.847**
IPLocationSet	0.96	**0.69**	**0.80**	301	72.75

5.3. Feedback Assessment

We conducted a user study on the quality of the feedback itself. We randomly selected 408 stitched frames with their feedback and showed them to eight judges (seven males and a female in their 20's), and asked them to evaluate the feedback that was automatically generated. We instructed them that the feedback goal is to decrease output video overlap, but not too much to the case that the overlap is not enough to stitch the two videos. Then, for every feedback frame, they needed to assess whether there was a need for feedback or not and in case the feedback was given if it was right or not. We report scores as percentages of the total set of the 408 frames using both average and standard deviation over the eight judges. To clarify, we compute the rate of correct feedback over the 408 frames for a judge j (let's call this R_j) and then report average and standard deviation of the set R_l for $1 \leq l \leq 8$. We use the notation $\mu(\pm\sigma)$ to denote the average and standard deviation respectively.

Out of the whole sample, users judged that the feedback success rate is 74.4($\pm$19)% which is computed by summing up cases where feedback is given and it is the right decision made by the system as well as cases where no feedback is given and that is again the right choice to undertake. Our results also indicate that the system is much more reliable in terms of signaling when feedback is needed as opposed to the case where it signals that no feedback is necessary.

5.4. Feedback Presentation

Our final experiment aimed at assessing feedback presentation and rate of generation, albeit on a small scale. We requested from three users (two males and one female in their 20's) who have tested the active feedback concept to evaluate the way the feedback was conveyed, whether through arrows, images or a combination of both.

While one of the users reported that the use of arrows was quite intuitive, two of them were complaining that arrows were sometimes confusing because they did not know how much they should move. On the other hand combining arrows with a feedback image was judged as a very useful presentation methodology, since the feedback images were helpful in understanding what the other user is shooting, and what is the best way to collaborate with him.

As for evaluating different rates for feedback generation, we have run experiments with three different rates (1 feedback / 0.5 second, 1 feedback / 1 second and 1 feedback / 3 seconds). The users who participated in this experiment judged that a feedback rate of once a second seems reasonable, while having it every 0.5 seconds is confusing and every 3 seconds is too slow.

6. CONCLUSION

In this paper, we introduced the concept of Active Feedback which provides guiding information for capturing users to maximize the probability of a successful panoramic video result. The system receives incoming user streams, analyzes them, generates the feedback, and sends it back to the shooting user in real-time to help him improving the live generated video. We discussed different cases where feedback can be provided along with triggers and presentation methods. For all of these aspects, we have conducted a user study aiming at making an intelligent choice for the design alternatives. Results show that adding the feedback component enhances the overall viewing experience measured by a number of different measures such as stitching precision and recall and output video size. As far as the future work is concerned, there are a number of areas worth investigating. First, providing feedback in a 3D manner (in all directions) can open new possibilities and obviously will face extra challenges such as accurate and fast 3D motion estimation. Second, it is worth investigating in how to maintain a video size that is somehow stable across frames and does not change abruptly as this was one of the desired behaviors gathered by the user study. Finally, it would be interesting to devise a scheme that can adaptively change the feedback rate based on device capabilities and network conditions.

7. REFERENCES

[1] “Live cast,” http://www.livecast.com/.

[2] “Qik,” http://www.qik.com/.

[3] A. Kaheel, M. El-Saban, M. Refaat, and M. Izz, “Mobicast - a system for collaborative event casting using mobile phones,” *ACM Mobile and Ubiquitous Multimedia - MUM*, Nov. 2009.

[4] N. Sorek and O. Bregman-Amitai, “Method for constructing a composite image,” *Samsung Electronics patent*, Jan. 2009.

[5] J. Hannuksela, P. Sangi, J. Heikkil a, X. Liu, and D. Doermann, “Document image mosaicing with mobile phones,” *ICIAP*, 2007.

[6] Aseem Agarwala, Maneesh Agrawala, Michael Cohen, David Salesin, and Rick Szeliski, “Photographing long scenes with multi-viewpoint panoramas,” in *Proc. SIGGRAPH*, 2006, vol. 25, pp. 853–861.

[7] M. El-Saban, M. Refaat, A. Kaheel, and A. Abdul Hamid, “Stitching videos streamed by mobile phones in real-time,” *ACM MM*, 2009.

[8] T. Shimizu, A. Yoneyama, and Y. Takishima, “A fast video stitching method for motion-compensated frames in compressed video streams,” *International Conference on Consumer Electronics*, 2006.

[9] J. Kopf, M. Uyttendale, O. Deussen, and M. Cohen, “Capturing and viewing gigapixel images,” *SIGGRAPH*, 2007.

[10] M. Brown and D. Lowe, “Automatic panoramic image stitching using invariant features,” *ICCV*, 2007.

[11] Baudisch and P. et al, “Panoramic viewfinder: providing a real-time preview to help users avoid flaws in panoramic pictures,” *OZCHI*, 2005.

[12] Aseem Agarwala, Colin Zheng, Chris Pal, Maneesh Agrawala, Michael Cohen, Brian Curless, David Salesin, , and Rick Szeliski, “Panoramic video textures,” in *Proc. SIGGRAPH*, 2005, vol. 24, pp. 821–827.

[13] J. Shi and C. Tomasi, “Good features to track,” in *Proc. CVPR*, 1994.

[14] R. Szeliski, “Image alignment and stitching: A tutorial,” *Microsoft Research*, 2006, Technical Report, MSR-TR-2004-92.

[15] A. Zomet, A. Levin, and S. Peleg, “Seamless image stitching by minimizing false edges,” *IEEE Transactions in image processing*, 2006.

[16] A. Agarwala, “Efficient gradient-domain compositing using quadtrees,” *ACM Transactions on Graphics*, 2007.

[17] Z. Duan, K. Gopalan, and Y. Dong, “Push vs. pull: Implications of protocol design on controlling unwanted traffic,” in *Proc. USENIX SRUTI*, 2005.

[18] J. Bouguet, “Pyramidal implementation of the lucas-kanade feature tracker: description of the algorithm,” *Intel Research Labs*, 2000, Technical report, OpenCV Document.

[19] J. K. Aggarwal and N. Nandhakumar, “On the computation of motion from sequences of images-a review,” in *Proc. IEEE*, 1988, vol. 76, pp. 917–935.

[20] Baker S., Scharstein D., Lewis, J. P., Roth S., Black M., and Szeliski R., “A database and evaluation methodology for optical flow,” *ICCV*, 2007.

[21] J. Boutellier, O. Silvn, M. Tico, and L. Korhonen, “Objective evaluation of image mosaics,” *International Conference VISIGRAPH*, 2007.