Abstract
We present a detailed exploration of VideoHandles, a novel interaction technique to support rapid review of wearable video camera data by re-performing gestures as a search query. The availability of wearable video capture devices has led to a significant increase in activity logging across a range of domains. However, searching through and reviewing footage for data curation can be a laborious and painstaking process. We showcase the use of gestures as search queries to support review and navigation of video data. By exploring example action-camera footage across a range of activities, we propose two video data navigation styles using gestures: prospective gesture tagging and retrospective gesture searching. This paper builds on our earlier VideoHandles work, reporting details of our interaction design and implementation, and presenting two additional evaluations. We demonstrate that VideoHandles is a viable interaction technique, returning promising results in its prototype form.

Keywords: Gestures, Video Search, Wearable Camera, Action Camera

1. Introduction
Max, a marine zoologist, is performing a scuba dive to record some underwater footage. He mounts an action camera to his chest and starts recording. During the dive Max performs a variety of hand gestures for his buddy, indicating aquatic life of interest. On one occasion, he sees a trigger fish and performs a fish swimming gesture followed by a trigger mime. On another occasion, he sees a puffer fish and performs the gesture (fish swimming gesture followed by a two-handed mimicked inflation to indicate ‘puffer’) to his dive buddy so that she can identify it too. In total 90 minutes of footage was captured. Upon returning home, Max uploads the footage to his computer with the intention of revisiting some of the key moments. He performs a puffer fish gesture as a search query and VideoHandles produces the puffer fish footage as a top ranking result among other results that include the fish swimming gesture - a key component across a range of diving gestures. After watching the puffer fish footage, Max notices the trigger fish footage among the returned results and decides to review that footage as well.

A wide variety of users, from amateurs to professionals, have adopted action cameras across a diverse range of activities, from mountain biking and scuba diving through to professional fieldwork. These cameras, such as the GoPro (www.gopro.com), are frequently mounted on head-gear or fixed to the chest and record throughout an activity, often for 1 - 2 hours, with little or no additional interaction. From these positions, and given a wide field of view (circa 170 degrees), the cameras are able to catch the majority of the wearer’s view including any interactions or gestures they may be performing with their hands. Where professionals may capture footage in order to maintain a clear record of their actions, others (e.g. sports enthusiasts) are more likely capture footage for key exciting moments.

Although these are different motivations, all scenarios necessitate review in order to locate desired moments. The most widely adopted current method for video review is video scrubbing (clicking through the timeline). However, as videos increase in length this process becomes inefficient and inaccurate [1].

We previously presented VideoHandles [2], a novel video search technique to expedite the review of specific moments in wearable video camera data. Our technique exploits action-camera’s wide field of view to capture the wearer’s interactions and gestures. VideoHandles allows users to query their footage by repeating interactions and gestures they performed during capture. As in our described scenario, these reproduced gestures are matched to instances in the original footage (Figure 1).

Based on observations of footage across a range of activities, we propose two video data navigation styles using gestures: prospective gesture tagging, where gestures are specifically performed to create a search marker, and retrospective gesture searching, where gestures are simply a part of the activity, recalled through muscle memory.

As an extension of our previous work [2], we present a more detailed exploration of the related work and our observations of a corpus of action-camera footage. Additionally, we provide further details of our prototype system design and present two additional lab-studies alongside our pilot in-the-wild study.

2. Related work
Our work builds on research in video navigation and search, and draws on research on gesture segmentation, gesture matching and memory.
2.1. Video Navigation

The most widely adopted method for video data navigation and review is video scrubbing. As a technique, it has been shown to be very fast in low latency systems where the video output updates perfectly in time with a moved slider [1]. Furthermore, user knowledge of the footage (temporal, contextual and spatial) helps to increase the speed of navigation [1]. However, scrubbing has its limitations. Firstly, any increase in latency has a linearly negative impact on scrubbing navigation time. Matejka et al. [1] report a 92% increase in navigation time given a latency increase of only 20 ms. While they were specifically concerned with internet streaming-related delays, this level of latency is not uncommon when locally reviewing high-resolution video on low-powered devices. Secondly, the mapping between the scrubbing slider and the corresponding video timeline is rarely one-to-one [3]. The slider is limited in size by the window width, itself defined by the video’s resolution. As the video length increases, each pixel of slider movement corresponds to a longer time step within the video. For example, given a 2.5 hour 1080p recording at full resolution (with a 1920 pixels wide slider bar), each pixel of slider movement corresponds to 4.69 seconds. Assuming the video is viewed at full resolution on a 24-inch 1080p monitor, a pixel measures 0.28 mm (or significantly less as resolution continues to increase). Moving 1 cm along the slider thus represents a time step of 2 minutes 47 seconds. These time steps make it easy to miss interesting moments of footage even with small slider movements.

Previous work has explored more efficient video visualization techniques to assist with navigation. Panopticon [4] provides a rapid overview of an entire video by displaying multiple sub-sequences in parallel. Similarly, Swifter [5] provides a series of thumbnails during video scrubbing that allows the user to gain a better understanding of the footage as a whole. While these techniques have been shown to improve user navigation efficiency, they rely on scene variation within the footage to assist navigation. As action-camera footage typically consists of one continuous shot of a single activity, it does not include this variation, thus making the selection of a specific moment from a thumbnail more difficult. Furthermore, these techniques were designed for navigating single video clips. The VideoHandles technique supports navigation and searching through video collections. VideoHandles aims to assist video navigation by providing location cues based on user’s input gesture. In this way, VideoHandles not only provides navigation cues, but also functions as a video search interface.

2.2. Video Search

Traditionally, in an attempt to align with other online search forms, video search has been based on a query-by-text approach [6]. This technique requires pre-defined textual annotations that necessitate initial time expenditure typically deemed unacceptable for large volumes of video [7]. Not only is time a limiting factor within this approach, but the annotations rely heavily on subjective perception, resulting in skewed emphasis towards user interpretation and making the sharing of annotations between multiple users difficult [8]. More recently, video search has begun to focus on a number of different approaches such as combining visual and audio cues [9], concept search [10], image search [11] and a combination of text, visual and concept [12]. While the limitations affecting these approaches differ, they are all challenged to some extent by a semantic gap, where user-associated high-level concepts and subjective meanings are not detectable by the computer [8]. Although video search results are improving, the best results are often achieved using a combination of image processing and human interaction [8].

2.3. Gesture Segmentation and Matching

As gestural interfaces have grown in popularity, so too has research on image processing, specifically gesture segmentation and matching. The approaches typically vary based on camera type [13] or key algorithmic features e.g. [14], [15]. More recently, approaches such as those presented by Song et al., demonstrate the continued effort and potential results of work in this area [16]. As yet there is no one-size-fits-all solution and the design decisions of any proposed algorithm need to be closely tied to that of the setting.

2.4. Gestures and Memory

We aim to exploit the positive effect that gestures have on our memory [17]. In order for this interaction technique to be
viable, we assume that at moments of special interest, users not only attend to the subject of interest, but also the actions they are performing. While in everyday usage this may not be universally true, the literature on gesture memorability supports our assumption. Kühnel et al. [18] found that the correspondence between gestures and their associated actions aided user’s memory. Nacenta et al. [19] found that people were able to best recall their own, self-defined gestures. VideoHandles draws on both of these points, enabling users to search using gestures from a known domain that they have previously defined and performed themselves during their activities.

3. An Exploration of Action Camera Footage

In order to gain a more detailed understanding of the typical usage of action and self-capture cameras, we observed and analyzed footage captured from a range of activities. In total we collated more than 50 hours of footage from a range of activities intended to represent a varied sample of those in which action-cameras are used. Our activities included snowboarding, power boating, tennis, cycling, hiking, running, scuba diving, windsurfing and archaeological excavation. The footage was collated from 5 existing users of this type of camera.

While the scenarios were very different, they are all examples of activities within skillful domains and provide a broad overview of the styles of footage captured and, more importantly for us, the kinds of actions performed during capture. We did not conduct a rigorous ethnographic study, rather provide lightweight motivating observations to help inform the design of our VideoHandles technique.

3.1. Observation 1: Activity-based Gestures

Our first observation is that the style of gestures naturally performed varies significantly according to the usage scenario. For example at one end of the spectrum, scuba diving includes a significant amount of sign-language gestures, where meaning is directly encoded in the hand-shape and motion. Due to the constraints of the environment, gestures occur frequently as all communication takes place in symbolic gestural form.

In the middle of the spectrum are activities such as archaeological excavation, tennis and windsurfing. Whilst these activities do not have a clear gestural vocabulary (like scuba diving), the majority of the skill is performed manually (i.e. embodied action performed specifically with the hands). For example, this could be careful trowel movements in archaeology, the various shots in tennis (forehand vs. backhand) or the different hand-holds for mast and boom positioning in windsurfing. Although the action is not bound to hand shape, the motion of the hands plays a key role, allowing these activities to also be good candidates for VideoHandles.

3.2. Observation 2: Non-Activity-based Gestures

At the other end of the spectrum, where manual variation is limited, are activities such as mountain biking and running. In these activities, the primary execution of the skill is non-hand based and thus the participant’s hands perform a limited variety of movements. For this reason, the ‘normal’ opportunities for gesture or action repetition are more limited without the performance of additional deliberate gestures, used as prospective markers for later searching.

Even during those activities whose skill is less manually-performed, one key similarity observed between all the activities is the frequent and continued use of social or ‘pantomime’ gestures. These settings showcase frequent language-like gestures, such as congratulatory ‘high-fives’ or ‘fist-bumps.’ These same activities also utilized language-tied and deictic gestures, such as ”that time you went left and I went the other way.”

3.3. Observation 3: Capture Purpose - Activity Capture vs. Footage Capture

Our final observation of action camera usage concerns capture style and the ‘purposefulness’ of shot framing. As previously mentioned in the introduction, action cameras for the most part are designed with large fields of view. This affords the opportunity to mount the camera to the desired location and initiate recording prior to beginning the chosen activity, safe in the knowledge that anything occurring in front of the wearer is likely to be captured. This ‘static’ usage style is typically adopted by those mostly concerned with ease of capture, capturing for review, or specifically capturing their perspective of an activity.

Juxtaposed to this style of usage are the more framing-centric approaches to capture. In these instances, the wearer may actively reposition the camera during filming to best frame the subject of the footage, performing traditional filming techniques such as ‘panning’. A well-used example of this within action camera filming is the mounting of the camera on the end of a pole, such that better framed selfie-shots can be captured. The focus of this style of footage may not be the activity per se, but the footage itself. Within this usage scenario, there is an obvious attention towards the final captured material.

4. Video Handles

VideoHandles is a video search technique based on reproduction of gestures. Our technique enables users to remember, or specifically plan, gestures produced during recording and to reproduce these gestures as search criteria to relocate specific moments in footage. Our technique reduces the requirement for human time and effort in reviewing vast reams of video data. Further to this, our technique supports wider exploration and comparison of footage by returning a range of results.
Action cameras are designed to capture a wide field of view, encompassing any actions or gestures that take place. Users can search this footage by repeating these actions / gestures from a similar viewpoint. By not making any assumptions about the style of gestures performed or their meaning, our technique can support a wide variety of gestures, including sign-language like scuba-diving gestures and manual skill based actions, such as trowelling in archaeology.

Based on our observations in the previous section, we propose two video data navigation styles using gestures: prospective tagging and retrospective searching.

4.1. Navigation Style 1 - Prospective Tagging

As users become accustomed to VideoHandles, gaining an understanding of the gestures that are matched most successfully by the system, we foresee increased performance of gestures during recording specifically designed for later retrieval. We term these gestures Prospective Tagging. For example, if a moment of immediate interest occurs during mountain biking, the rider could pre-emptively perform a gesture to support and increase the accuracy of later retrieval (e.g. a gesture that would not normally occur during the activity). Furthermore, users may develop entire vocabularies, grouping activities on-the-fly with prospective gestures. Prospective marking of this kind further aids users memory of the gesture for search, thus ensuring more accurate search results.

4.2. Navigation Style 2 - Retrospective Search

VideoHandles is also able to support occasions where gestures are simply a part of the activity. In some instances, prospective marking will not be possible, perhaps because the event only acquires importance and meaning for the user in retrospect rather than at the time, or because the user is not interested only in a single event, but wishes to review and compare all examples of a particular activity. In these cases, the user can apply VideoHandles to perform a Retrospective Search. One of the benefits of this search method is that the user may be able to rely on visual and muscle memory to perform the query. If the search is for one or more instances of a well-rehearsed manual skill, then the previous practice will also enhance the consistency of the search query.

4.3. Multiple Results

Just as a web search provides multiple results, VideoHandles does not intend to provide only exact matches; rather it supports reflection and comparison between hits by returning a ranked range of results. In this way, VideoHandles also serves to better support chance finds when clicking through footage.

5. Prototype Algorithm

We developed a prototype system to explore the feasibility of our concept and its value from an HCI perspective. We used a combination of existing computer vision algorithms to track, segment, and shape- and motion-match gestures in different videos. The technical approach we adopt is just one of many possible approaches and any appropriate computer vision algorithm could be used. We detail our technical implementation here to better situate the results we present in our later studies.

We developed our system using the opencv framework. The intention of our system was not to focus on efficiency and speed of results, rather to explore the combinatorial requirements of a system that would be robust to variations in lighting, scene clutter and background changes. As such we developed our prototype to be used offline. In its current form, our algorithm takes on average 68% of total clip time to process results.

5.1. Segmentation

The accurate segmentation of data from fast moving, noisy footage is a well-documented problem. Significant variation in lighting, background and scene clutter necessitate a robust approach. During the analysis of our sample footage we found that the majority of gestures performed were highly dynamic, taking place over a relatively short period of time and making use of both motion and shape for emphasis and meaning. Given this observation, we segment gestural information from the scene based on motion, through Farnebäck optical flow [20]. From these motion regions, skin candidate areas are found using a combination of RGB and HSV skin color detection.

$$
rgb\text{skin} = \begin{cases} 
    r > 95, g > 40, b > 20 \\
    r - g > 15, r > g, r > b \\
\end{cases}
$$

$$
hsv\text{skin} = \begin{cases} 
    h > 10, b < 244 \\
    s > 64 \\
    \text{GaussianBlur}(h * s) > 200 \\
\end{cases}
$$

The motion and color segmentation returns regions of probable skin. These regions are ignored if they do not fit within certain size criteria. As we are focusing on head and chest mounted cameras, relatively accurate thresholds can be calibrated based on average reach and average hand size.

5.2. Tracking

Once we have identified probable skin regions within our video frame, we then perform tracking. Once a skin region has been successfully tracked over ten frames, its details (starting shape, end shape and path) are saved as a gesture chunk. When gesture chunks have been located within the query footage, they are compared with the raw footage to determine suitable matches.

5.3. Chunk Comparison

The comparison considers the similarity of the chunks’ shape and motion. The shape matching is conducted using Fourier Descriptors of the contours and chamfer matching [21]. Both of our adopted matching techniques are rotation and scale invariant. By varying the gesture chunk window size in the raw footage our techniques are also time invariant, taking into account performances of the same gesture with different temporal properties. The motion matching is performed using the $S$ gesture recognizer [22] (as we treat our gestures as 2D within the
camera view plane. Similarly to our shape matching, this is invariant to gestures performed over different amounts of time. It is also invariant to scale and location.

5.4. Scoring

Once all the chunks have been compared, temporally and spatially co-located matched chunks are grouped together into gestures in the raw footage. These gestures can then be compared in order to return ranked search results. Gestures are ranked according to: the total number of matched chunks, average path match scores and average shape match scores.

6. Examining the Performance of Our Algorithm

Having developed a prototype to enable further exploration of the VideoHandles technique, we were keen to explore its accuracy through a range of different studies. As previously discussed, self-captured video is typically noisy, subject to fast changes in motion, constant vibration, and captured in dynamic lighting. For this reason, a key challenge in the automatic processing of footage of this kind is the reliable segmentation of features of interest. In study 1, we explore the ability of our algorithm to segment the wearer’s skin regions across a range of footage. Secondly, our interaction technique relies on the ability to successfully compare and match instances of the same gestures. We explore this in our second study.

6.1. Study 1: Assessing our Prototype’s Segmentation Accuracy

Color-based skin segmentation in RGB and HSV space is known to be imperfect under lab test conditions [23]. In this study we explore whether the approach can be good enough to facilitate segmentation for gesture matching in the wild. In this instance, we allow for false-positives (non-skin regions identified as skin), instead focusing on the correct identification of skin regions.

We selected 25 frames from over 40 hours of collated footage. The frames were ‘cherry-picked’ for their variation in lighting and background clutter. We recruited 5 participants to identify and outline any skin pixels in the frames. As our algorithm uses optical flow to perform image segmentation, we designed our study to allow the human participants to also use any knowledge gained from the previous frames. Therefore, the participants were shown 3 seconds of footage, culminating in the selected frame. The participants were instructed to draw around any skin pixels in the last frame using a mouse and a purpose-built interface. The participants each processed all of the frames and their results were compared and accumulated into the minimum agreed regions (a binary ‘and’ was performed across all the output images).

Next we processed the 25 clips through our system. Similarly to the human participant experiment, the system was fed the preceding three seconds of footage, culminating in the desired frame. As in the human participant case, the system output a binary image showing any skin pixels. We compared the algorithm’s recognized skin regions against the human identified regions. We measured the proportion of the human identified region that the computer also identified.

![Figure 3: Image showing output skin regions from participants (blue) and algorithm (green) on top of cherry-picked video frames showcasing a range of different backgrounds and lighting.]

6.2. Results

Our skin detection algorithm found 80% of the skin regions identified by the human participants. That is, a sufficient number of skin pixels were located for the region to be treated as a hand candidate. On a per-pixel level our algorithm found on average 60% of the identified skin pixels. In only 8% of the images where the participants identified skin regions, did the algorithm return no results (as in the underwater image in figure 3).

While far from a perfect score, we suggest that this result is sufficient and practicable to support our algorithm for a number of reasons. Firstly, our algorithm relies on motion data to segment the image prior to skin detection. This results in only pixels of dynamic color change (typically edge pixels) being available for further selection. As visible in Figure 3, where our color segmentation is imperfect, it tends to result in the selection of edge pixels, still enabling accurate contour extraction and, therefore, shape comparison. Secondly, our participants defined skin regions using an average contour drawing technique. This resulted in selections larger than the exact skin regions. If the participant’s selections were performed using a technique allowing for finer accuracy, then our average detection score would increase.

You can see from the example output images that our color segmentation identifies a range of false-positives alongside any correct matches, a frequent occurrence with color segmentation techniques. These false-positive regions, if larger than a given threshold, are also treated as potential gestures by our system and thus are matched against other skin-regions for similarity. Whilst occasional ‘noise’ matches may be found, as we provide a range of matches as opposed to a sole ‘best’ match, this does not have a large negative impact on the output of any results. The example comparison frames above also showcase the effect of environmental variables on our segmentation technique.

This is especially visible in the underwater example, where the additional blue hue from the water results in no skin color detection.
Participants were asked to produce gestures based on a gesture stimulus on video. These individual clips were used as stimulus as closely as possible. These data were also recorded and participants were asked to reproduce their gesture for that stimulus. Different stimuli were re-displayed one at a time in a random order for a subsequent participant.

These gestures were recorded in video clips and were later arranged into 16 unique sets of 4-videos (i.e. each set consisted of four gesture video clips and no two of the 16 sets were the same). These sets were used as raw data for a subsequent participant.

Task 2 - Participants recorded a gesture query. The previous stimuli were re-displayed one at a time in a random order and participants were asked to reproduce their gesture for that stimulus as closely as possible. These data were also recorded on video. These individual clips were used as query data for a subsequent participant.

Task 3 - The participants were shown a query clip and asked to locate the closest matching clip within a raw data set (recorded by a previous participant). They were asked to identify any potential gesture matches and provide a match score on a continuous scale between 0 and 100. Using a different participant to rank the similarity of the two instances allowed us to remove the semantic meaning of the gesture as a variable from the test, leaving only the similarity of its appearance as a factor.

We alternated stimuli so that participants performed gestures based on different cues from those they had matched (Figure 4 shows one of two stimuli sets). We also counterbalanced the task order against the possibility of learning effects, so for half of the participants the tasks were performed in the order described above and for the other half, task 3 was performed prior to tasks 1 and 2. As each participant used the previous participant’s gestures for task 3, participant 1 returned at the end when data was available for them to perform task 3. We then ran our algorithm over the same data set, which also returns ranked search results (ranked 1-4, or – if not identified).

We compared this result to the results provided by our human participants, and aligned the rankings if more than one matching gesture had been identified by them.

6.4. Results

Our prototype algorithm returned the same preferred gesture (best match) as our human study participants 43.4% of the time. However, when the systems top two ranked results are considered, then our system returned the correct match 75.5% of the time. From the five possible results (including ‘no match’) our algorithm’s average position of the participant’s matched result was 1.65 (i.e. the participant’s matched gesture fell between the first two results of the computer matched gesture). As we are trying to reduce the overall time in which the gesture is found, the ranking of the result is not as important as a matched gesture being identified correctly within a limited range of results. Overall, 89% of gestures were correctly identified (within the four ranked results). In the context of existing video search techniques, this means that in around 9 out of 10 cases gestures could be found rapidly by discretely jumping to the time before any linear searching was required.

Our participant’s depicted information on the cue cards through a mixture of both shape-rich and motion-rich gestures (Figure 4). The motion-rich gestures were typically performed using a pointing gesture. In total, we classified 60% of the gestures performed as motion-intensive and 40% as shape-intensive.
Interestingly, the participants performing a larger portion of shape-intensive gestures typically received a higher match accuracy.

7. Pilot Study of VideoHandles in the Wild

Figure 5: Example frames from the ‘in the wild’ footage. Left: ‘two-finger gun’ gesture denoting red cars. The video frame is subject to natural highlights and shadows. Center: ‘OK’ gesture denoting feeling energetic. Due to inexperience with action cameras, P1 frequently performed gestures at the extremity of the camera’s field of view, resulting in lost ‘gesture information’. Right: ‘wave’ gesture denoting feeling tired. As visible in the still frame, the gesture has very low ‘shape resolution’, resulting in less accurate matches in our prototype.

To further explore our interaction technique and to begin to evaluate our prototype, we conducted a pilot study of VideoHandles in realistic use. Two participants wore GoPro action cameras on chest mounts whilst cycling, recording 45 minutes of footage on average. The participants were asked to perform a gesture of their choosing, indicating: every time they saw a red car, when they were feeling energetic and when they felt tired. Examples of participant 1’s (P1) gestures can be seen in Figure 5.

After the activity was completed, P1 reviewed their footage using video scrubbing, recording the time and meaning of every gesture they saw. It is worth noting that through this approach some gestures may have been skipped-over (missed), but a subset were located to provide correspondences for our VideoHandles software. In total, P1 identified 28 gestures. After review, an example of each type of gesture was recorded as a search query for our system. P2 was excluded from the study due to the similarity of the gestures they performed; a ‘thumbs up’ indicating energetic and a ‘thumbs down’ indicating tired. Due to the rotation invariance of our gesture matching technique, our prototype cannot differentiate between these gestures in its current form.

7.1. Results

P1 indicated 28 ‘two-finger gun’ style gestures corresponding to red cars. Our prototype algorithm returned 89% of these gestures and 1 false-positive gesture. P1 performed 3 ‘OK’ gestures and our prototype algorithm returned 2 correct matches and 16 false-positives gestures, including both car gestures and tired gestures. P1 performed 2 ‘tired’ gestures which our prototype did not match, but produced 8 false-positive gestures.

8. Discussion

Our results showcase the promise of our prototype algorithm and highlight the potential of our interaction technique as a whole. We demonstrate 80% hand detection, 75.5% hand matching in our lab study, and 89% gesture matching across 28 gestures performed in the wild. Our exploration and evaluation of our approach has highlighted a number of interesting features of our design.

Firstly, the majority of action-cam devices have a wide field-of-view (typically between 160 and 170 degrees). While this can lead to some radial distortion towards the edge of the frame, it serves well to capture the actions performed in a wide area in front of it. Even given a wide field-of-view, shot framing is still an important function of their use and subject to human error, specifically amongst inexperienced users. For example, in our study 3, our participant repeatedly performed their ‘OK’ gesture at the very top of the frame, resulting in part of the shape detail being lost. This had a negative impact on VideoHandles ability to correctly identify and match the gesture. Given increasing use of these cameras and our technique, we foresee users becoming more accustomed to the camera’s field of view, thus positioning more accurately.

Secondly, our studies highlighted differences between human and computer interpretation of gestures. As our gesture matching technique is rotation invariant, gestures that have significantly different meaning for humans are interpreted identically by the algorithm, such as thumbs up and thumbs down. Alongside the importance of the gesture’s rotation invariance, is the requirement for a unique contour. For example, the outer shape of the thumbs up gesture and the ‘OK’ gesture are very similar. To this end, our technique benefits from gestures of an unnatural shape, that are contour and rotation unique, rather than human-meaning unique. As a direct consequence of this requirement, a second participant was not used for study 3 as they used thumbs up and thumbs down gestures across two conditions, not providing a unique match.

As a result of the uniqueness requirements of our algorithm and our use of a ‘wide-fit’ skin color model, our prototype returns a number of false-positives and false-negatives. In study 3, our algorithm returned 3 minutes of false positive gestures within 42 minutes of footage. Even should this additional footage be watched without additional video scrubbing being used, it would still represent a significant time saving over watching the entirety of the original footage. Furthermore, by returning additional false results, our algorithm supports wider exploration of the footage, similar to the serendipitous results returned by a Google Web search.

While our results show significant promise for our interaction technique, we did not test extreme lighting conditions. As one can imagine, this would have a negative impact on the success and accuracy of our segmentation. Further to this, as lighting decreases, the output quality of action-cameras deteriorates significantly and color, contrast and saturation detail is lost. In this scenario, color segmentation of any kind would have little success. However, action cameras are designed to operate without a requirement for interaction and thus perform automatic white-balancing and contrast adjustments. This serves to maintain a more uniform brightness throughout the footage, reducing glare and maintaining shadow detail. This further aids our color segmentation approach.
9. Conclusion

**VideoHandles** is a novel search interaction technique for action camera footage which allows users to search through footage by repeating actions performed during the original recording. **VideoHandles** allows real-time tagging and categorizing of data, thus reducing time spent on post-processing, whilst facilitating wider exploration of recorded footage by supporting comparison between search matches. Our technique also supports a range of usage methods, allowing for both retrospective searching through memory of actions or prospective tagging of footage with specific gestures during the initial capture.

In addition to our original work [2] we have provided a detailed exploration of the style of footage captured by the action-camera community, described in more detail the implementation of our prototype system and explored the feasibility of its use in two lab studies and one study in the wild. Our results show 75.5% of correct matches returned within the first 2 results in our lab study, and 80% accurate hand detection across a range of environments. Alongside this, we also demonstrate promising initial results in the wild.

10. Future Work

Our work here has been an early exploration of the feasibility of using gestures and actions as search criteria for video data. There is significant opportunity for future work in this space.

Our prototype system relies heavily upon, and is constrained by, the skin-color segmentation of the frame. Our current implementation uses a best-guess, widest-fit skin color model for matching, designed to provide acceptably accurate results across the broadest range of possible users. In a final implementation, this segmentation could be extended such that it produces more accurate results. For example, users could ‘train’ the system to their skin color by indicating a region of target color in an example frame. By selecting an example frame from the raw footage, the skin color could be specified per the target lighting condition, improving the accuracy of the results. Further to this, this method would allow for the selection of other non-skin color sections. By selecting non-skin colors, the system could then account for users clothing etc. In this scenario, annotating the hands / gloves with unusual colours could also lead to significantly improved results.

Our technique currently supports querying based on the gestures of the camera wearer. However, action-cameras are often used during activities with multiple participants. As such, an interesting extension of our technique would be to support the repetition of any gesture that takes place within the frame and not only that of the wearer. This, in turn, would allow for a more complex range of group queries, results and reflection.

References


