Toward Crowdsourcing Micro-Level Behavior Annotations: The Challenges of Interface, Training, and Generalization

Sunghyun Park, Philippa Shoemark, and Louis-Philippe Morency
Institute for Creative Technologies
University of Southern California
12015 Waterfront Dr., Los Angeles, CA 90094
{park, morency}@ict.usc.edu, pjshoemark@gmail.com

ABSTRACT

Research that involves human behavior analysis usually requires laborious and costly efforts for obtaining micro-level behavior annotations on a large video corpus. With the emerging paradigm of crowdsourcing however, these efforts can be considerably reduced. We first present OCTAB (Online Crowdsourcing Tool for Annotations of Behaviors), a web-based annotation tool that allows precise and convenient behavior annotations in videos, directly portable to popular crowdsourcing platforms. As part of OCTAB, we introduce a training module with specialized visualizations. The training module's design was inspired by an observational study of local experienced coders, and it enables an iterative procedure for effectively training crowd workers online. Finally, we present an extensive set of experiments that evaluates the feasibility of our crowdsourcing approach for obtaining micro-level behavior annotations in videos, showing the reliability improvement in annotation accuracy when properly training online crowd workers. We also show the generalization of our training approach to a new independent video corpus.

Author Keywords
Crowdsourcing; micro-level annotations; behavior annotations; inter-rater reliability; training crowd workers

ACM Classification Keywords
H.5.2. Information interfaces and presentation: User Interfaces.

General Terms
Design; Experimentation; Human Factors; Measurement.

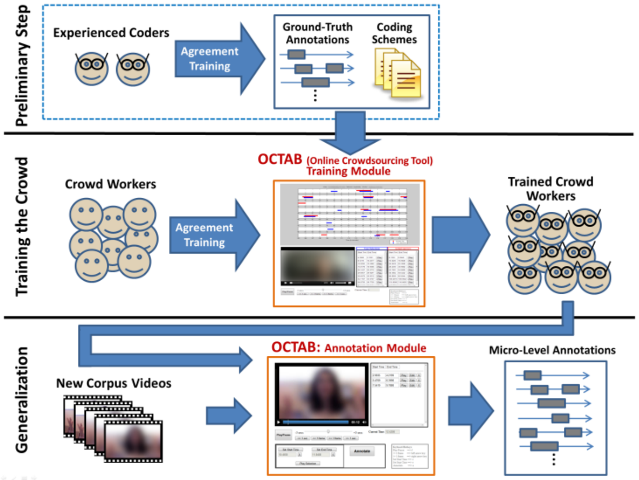


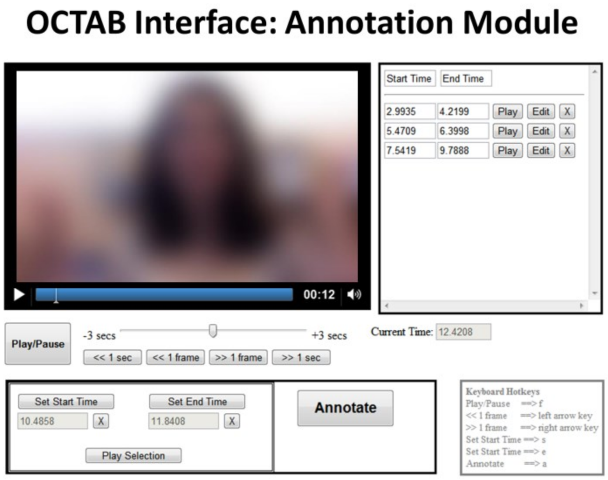
Figure 1. Overview of our approach for crowdsourcing micro-level behavior annotations in videos, with a focus on our web interface called OCTAB, which (1) includes a module specifically designed to train crowd workers online and (2) generalizes to new independent video corpora.

INTRODUCTION

Annotating multimedia content is becoming an important part of many recent research problems, including multimedia event recognition [21], video retrieval and classification [13], and human behavior analysis [19]. Supervised learning approaches applied to these research problems usually require a large number of annotated video sequences. While some of these algorithms are applied at the video or scene level (referred to as macro-level annotations), many of these problems need micro-level annotations, where the precise start and end times of an event or behavior need to be identified. These annotation efforts, which are usually carried out with experienced local coders, are very costly both in terms of budget and time.

In recent years, there has been an explosive growth in the research and use of crowdsourcing paradigm, fueled by convenient online crowdsourcing environments like Amazon Mechanical Turk. In the research community, crowdsourcing is already being actively used for many types of tasks, including image labeling [18] and linguistic annotations [17]. When crowdsourcing micro-level human behavior annotations in videos, three main challenges emerge: interface, training crowd workers online, and generalization. Firstly, there is a need of a web interface that allows crowd workers to accurately and efficiently annotate micro-level behavioral events while keeping the interface simple and intuitive. Secondly, there should be an effective web interface and procedure for training crowd workers online that can simulate the environment experienced local coders use when discussing and reaching

IUI'14, February 24–27, 2014, Haifa, Israel.
Copyright © 2014 ACM 978-1-4503-2184-6/14/02...$15.00.
http://dx.doi.org/10.1145/2557500.2557512



OCTAB Interface: Annotation Module

Start Time	End Time
2.9935	4.2199
5.4709	6.3998
7.5419	9.7888

Current Time: 12.4208

Set Start Time: 10.4858

Set End Time: 11.8408

Keyboard Hotkeys

Play/Pause	==> f
<< 1 frame	==> left arrow key
>> 1 frame	==> right arrow key
Set Start Time	==> s
Set Start Time	==> e
Annotate	==> a

Figure 2. The first component of OCTAB (Online Crowdsourcing Tool for Annotations of Behaviors) is a web annotation module that allows crowd workers to make precise micro-level annotations of human behaviors or events in videos.

agreement. Lastly, the training of online workers should generalize across datasets if we want this approach to be widely applicable.

In this paper, we present OCTAB (Online Crowdsourcing Tool for Annotations of Behaviors), a web-based annotation tool that allows precise and convenient behavior annotations in videos, directly portable to popular crowdsourcing platforms such as Amazon Mechanical Turk (see Figure 1). In addition, we introduce a training module with specialized visualizations and an iterative procedure for effectively training crowd workers online, inspired by an observational study of experienced local coders reaching agreement. Finally, we present an extensive set of experiments that evaluates the feasibility of our crowdsourcing approach for obtaining micro-level behavior annotations in videos, showing the reliability improvement in annotation accuracy when properly training online crowd workers. We also show the generalization of our training approach to a new independent corpus.

RELATED WORK

Crowdsourcing has gained much attention lately, and a survey paper by Yuen et al. [31] and another by Quinn and Bederson [22] present a general overview of the topics on crowdsourcing and human computation, and many interesting applications [2, 9, 32] are appearing that take advantage of the new paradigm. Regarding Amazon Mechanical Turk, Mason and Suri [15] provided detailed explanations on using the platform for conducting behavioral research, and Ross et al. [25] showed changing demographics of the people using the platform.

Quality control is a critical issue with crowdsourcing. Downs et al. [5] and Rashtchian et al. [23] showed the benefit of a screening/qualification process, Le et al. [12] showed an approach of adding a training period in designing a study, and Sheng et al. [26] explored repeated labeling of data for more reliability. By comparing annotations (none of them on videos) obtained with crowdsourcing and those with expert annotators, several [6, 7, 14, 18, 23, 27] have reported across different domains that they could obtain good quality annotations through crowdsourcing. In our work, we incorporate most of these quality control measures and further show novel experimental results of comparing micro-level annotations in videos obtained by crowdsourcing with those done by experienced local annotators.

As for crowdsourcing video-related tasks, Wu et al. [30] worked on obtaining video summarizations, Biel and Gatica-Perez [3] on macro-labeling impressions of vloggers in videos, and Riek et al. [24] on macro-labeling social contexts in video scenes. However, none of them were concerned with micro-level annotations. Probably most relevant pieces of work in terms of our web interface were done by Vondrick et al. [29] and Spiro et al. [28], whose interfaces allowed micro-level motion tracking and were also used with Amazon Mechanical Turk. However, their interfaces only put an emphasis on motion tracking, while our interface is concerned with identifying and segmenting behavioral events in videos. Although there are quite a number of software for making complicated annotations on videos [4], such full-fledged tools are not suitable to be used for crowdsourcing due to a relatively steep learning curve and the difficulty of incorporating them into web-based crowdsourcing platforms.

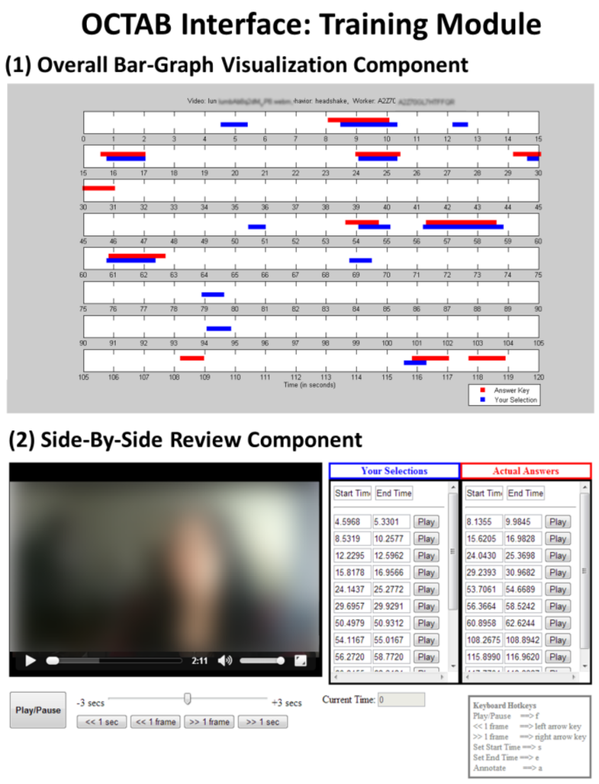
Krippendorff's alpha has been previously used to measure inter-rater reliability of video annotations both at a macro-level [24] (label on the whole video clip) and micro-level [8] . In this paper, we follow the approach taken in [8] at a micro-level, but we further explore the stability and reliability of the alpha at different temporal resolutions. The approach taken in [20] is also used for disagreement analysis to supplement the alpha because the alpha cannot show the types of disagreement between coders, which can be critical information for effectively training crowd workers.

To our knowledge, we are the first to introduce an effective interface with specialized visualizations to train crowd workers online, by extensively showing the feasibility of training crowd workers to obtain micro-level behavior annotations in videos and demonstrating generalizability of training across different video corpora.

ONLINE CROWDSOURCING TOOL FOR ANNOTATIONS OF BEHAVIORS (OCTAB)

We developed OCTAB[1] (Online Crowdsourcing Tool for Annotations of Behaviors), a web-based annotation tool for

[1] OCTAB will be made freely available for research at http://multicomp.ict.usc.edu/



Your Selections

Start Tim	End Time	
4.5968	5.3301	Play
8.5319	10.2577	Play
12.2295	12.5962	Play
15.8178	16.9566	Play
24.1437	25.2772	Play
29.6957	29.9291	Play
50.4979	50.9312	Play
54.1167	55.0167	Play
56.2720	58.7720	Play

Actual Answers

Start Tim	End Time	
8.1355	9.9845	Play
15.6205	16.9828	Play
24.0430	25.3698	Play
29.2393	30.9682	Play
53.7061	54.6689	Play
56.3664	58.5242	Play
60.8958	62.6244	Play
108.2675	108.8942	Play
115.8990	116.9620	Play

Current Time: 0

Keyboard Hotkeys
Play/Pause ==> f
<< 1 frame ==> left arrow key
>> 1 frame ==> right arrow key
Set Start Time ==> s
Set End Time ==> e
Annotate ==> a

Figure 3. The second module of OCTAB to effectively train crowd workers online by giving them a quick overall visualization of disagreement (top) and the ability to review both ground-truth and their attempted annotations side-by-side (bottom).

making convenient and precise micro-level annotations in videos. It consists of two main modules. The first module is a HTML-based web interface that allows an annotator to conveniently navigate in a video to annotate micro-level human behaviors or events (see Figure 2). The second module was designed for training crowd workers online, inspired by observing how experienced local coders train themselves to reach agreement (see Figure 3).

Annotation Module (Micro-Level Behavior Annotations)

OCTAB is intended for annotating a single behavior on a single video at a time, and it is based on HTML5 and JavaScript, providing all the basic functionalities of a web video player (HTML5 supports three video types of MP4, WebM, and Ogg). We considered the following three main aspects in our design of the annotation module of OCTAB.

Precision

For accurate micro-level annotations on videos, annotators need to have frame-level precision in identifying the start and end time points of an event. To address this requirement, the interface provides the annotator with 4 buttons for moving 1 second backward/forward and 1 frame backward/forward from the current time in the video, as well as a slider bar that offers frame-level navigation in the range from -3 to +3 seconds. Once the annotator identifies a behavior or event to annotate, he/she can use the navigation control buttons to pinpoint and select the behavior or event's start and end times. Then, he/she can play the selection to verify and press a button to save the selection as a valid annotation. Although intended for annotating a single behavior on a single video at a time, it should be noted that this interface also allows annotations of multiple behavior tiers or intensities with a simple addition of radio buttons, and it can be even configured to support any arbitrary annotation tasks with additional radio buttons, sliders, text boxes, etc.

Integrability

Popular annotation software applications like ELAN or ANVIL [4] allow annotators to make sophisticated annotations on video and audio files, but they are not suitable for the purpose of crowdsourcing. They have a relatively steep learning curve to use and cannot be used with online crowdsourcing platforms like Amazon Mechanical Turk. OCTAB was written directly in HTML so that it can be easily used to create a template task page when using online crowdsourcing platforms.

Usability

Annotating videos often involves moving around in a video to check, re-evaluate and edit previously made annotations. A special section in the annotation module displays a list of all saved annotations, and the annotator can always go back and work on previously made annotations by replaying, editing or deleting any annotations. For convenience and speed in making annotations, most controls in the interface have hotkeys associated with them, and the interface's functionalities are kept to the minimal level with an intuitive layout to minimize confusion.

Training Module (Training Crowd Workers Online)

The challenge of training crowd workers for annotation tasks arises mainly due to the lack of physical interaction that local coders enjoy when training themselves in person according to a coding scheme. In order to have an effective design of this training module, we first observed how experienced local coders work together to reach agreement. Then, we created needed visualizations and a training procedure to translate the findings to effectively train crowd workers online.

Observational Study of Experienced Local Coders' Training

As a preliminary step, we performed an observational study of two experienced local coders reaching agreement on behavior annotations for 5 short YouTube videos of people giving movie reviews. The coders annotated a total of 4 behaviors, the same ones used in our experiments: gaze away, pause filler, frown, and headshake (see Experiment section).

We observed that experienced local coders sit together to devise a coding scheme, which is a precise description of an annotation task. Then, they individually try annotating according to the coding scheme on a training video. After computing their agreement, they again sit head-to-head to review their annotations together, replay all of their annotations multiple times side by side, engage in discussions, and make appropriate modifications to the coding scheme as needed. This process is iterated with more training videos one after another until the agreement consistently reaches a satisfactory level determined by researchers.

From the observational study, we noted that our online training module should concentrate on two key functionalities in order to simulate how local coders train themselves. Firstly, crowd workers should have an overall visualization that enables them to quickly compare their annotations with each other (or with ground-truth annotations). Secondly, crowd workers should also be able to efficiently review (play the video and see all instances of) both ground-truth and their attempted annotations side-by-side.

Design of Online Training Modules

The first necessary functionality noted during the observational study is reflected in our training module with an overall bar-graph visualization on a time line that not only informs crowd workers with an overall picture of their mistakes in identification of a behavior but also in its segmentation (see Overall Bar-Graph Visualization Component from Figure 3). The second functionality is reflected with a modified version of the behavior annotation module in which crowd workers can review both ground-truth annotations and their attempted annotations side-by-side by repeatedly playing any of those annotation instances in the video (see Side-by-Side Review Component from Figure 3). This training module is generated automatically with scripts.

PROCEDURE FOR CROWDSOURCING MICRO-LEVEL BEHAVIOR ANNOTATIONS IN VIDEOS

Given our interactive web interface for training crowd workers and annotating micro-level behaviors in videos, we propose 4 main steps to successfully train new crowd workers: Obtaining coding schemes and ground-truth annotations, recruiting and screening workers, training workers online, and obtaining repeated annotations if necessary.

Obtaining Coding Schemes / Ground-Truth Annotations

If no trained online workers are available, the first step is to work with experienced local coders to create a coding scheme and annotating a small set of training videos. As will be shown, this step of creating a coding scheme with annotated training examples is only necessary if the behavior to annotate is new. During this step, the local coders train themselves on the training videos until their agreement reach a satisfactory level (see Experiments section for more details about agreement measures during their training sessions). The resulting annotations from these training videos will be used as ground-truth annotations for training crowd workers. If trained online workers are available for the desired behavior or if a coding scheme and annotated training set already exist, this step can be skipped.

Recruiting / Screening Crowd Workers

In recruiting crowd workers, it is suggested to first try recruiting from a forum such as www.mturk.com, where many serious crowd workers reside. It is also beneficial to use a relatively unambiguous annotation task that still requires close attention to detail at the frame level to check if a crowd worker is able to annotate with frame-level precision. For example, gaze away behavior is a relatively easy behavior to identify with unambiguous start and end times, but it requires one to pay attention at the frame level. Measuring agreement performance on this type of tasks can be a good threshold point for screening crowd workers.

Training Crowd Workers Online

For training crowd workers, we propose an iterative procedure where workers first annotate a video with OCTAB annotation module and then receive feedback with the training module. This gives them a chance to learn and improve with each training video using the overall bar-graph visualization and side-by-side review components. Once crowd workers consistently perform at the agreement level on par with the agreement between local coders, they are tagged as properly trained. For our study, we used Time-Slice Krippendorff's alpha (described in the Experiment section) to measure agreement, and we set the satisfactory alpha level at 0.80 for relatively clear behaviors and 0.70 for harder ones.

Unique vs. Repeated Annotations

When annotators are trained to strongly agree with each other, future annotations can be obtained with one annotator per video. With properly trained crowd workers, it could be the case that having only one worker annotate per video is sufficient to obtain quality annotations. However, for relatively harder behaviors to annotate, it may be necessary to make repeated annotations with multiple workers per video and take a majority vote approach. In fact, it could be possible to take this approach with even untrained crowd workers and obtain annotations with satisfactory quality. We show the effect of training and having repeated annotations with an extensive set of experiments in this paper.

EXPERIMENTS

We designed our experiments to evaluate the performance and user experience of our OCTAB interface for online crowd annotations. We particularly put a focus on the effect of training crowd workers and also tested the generalization of our training procedure by having workers trained on one dataset and have them tested on another independent dataset.

Evaluation Methods

We used Time-Slice Krippendorff's alpha [8] as our main evaluation metric for measuring inter-rater reliability of micro-level annotations in videos. Krippendorff's alpha is

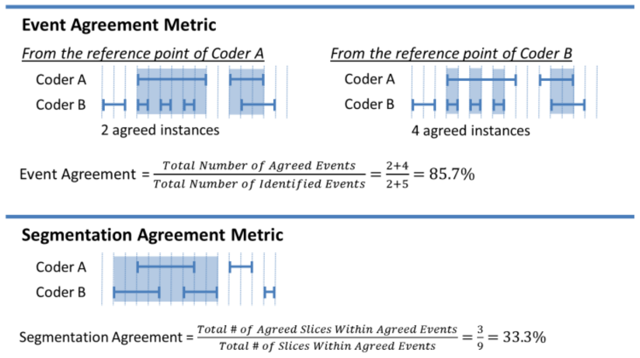

Event Agreement Metric
From the reference point of Coder A
Coder A
Coder B
2 agreed instances
From the reference point of Coder B
Coder A
Coder B
4 agreed instances
Event Agreement = Total Number of Agreed Events / Total Number of Identified Events = 2+4 / 2+5 = 85.7%
Segmentation Agreement Metric
Coder A
Coder B
Segmentation Agreement = Total # of Agreed Slices Within Agreed Events / Total # of Slices Within Agreed Events = 3/9 = 33.3%

Figure 4. Definition of the event and segmentation agreement metrics with examples.

particularly suited for crowdsourcing because it can handle multiple annotators at the same time and also account for missing data. We further used two supplementary metrics to analyze the types of disagreement between coders, which can be very helpful in determining whether coder disagreement stems from inaccurate identification of behaviors or from imprecise segmentation [20].

Time-Slice Krippendorff's Alpha

Our first measure, Krippendorff's alpha [10], is a generalized chance-corrected agreement coefficient that can be calculated between two or more annotators. The general formula for the alpha is the following:

$$\alpha = 1 - \frac{D_o}{D_e} \qquad (1)$$

where D_o, or observed disagreement, is the amount of pairwise disagreement observed between the annotators, and D_e, or expected disagreement, is the level of disagreement expected by chance as calculated from the data. The coefficient alpha itself is a measure of agreement ranging from -1 to 1, where 1 is perfect agreement (zero observed disagreement), 0 is chance-level agreement, and values lower than 0 indicate systematic disagreement.

The alpha works by looking separately at the agreement on individual annotation instances. For micro-level annotations, we treat each time slice (e.g., 1 frame per slice) as a separate annotation instance, with a binary annotation indicating presence or absence of a specific behavior (such as a frown). While it is the case that adjacent frames tend to have similar annotations, our experiments show that the alpha is not very sensitive to the sampling rate of the time slice. The agreement is calculated separately for each annotated behavior.

Applying the alpha to individual time slices means that the measure can only assess whether the annotators agree that at a certain time point a behavior takes place, not whether they agree about the segmentation or the individuation of behaviors (whether a certain time span contains one or two instances of a frown); this drawback has been pointed out by Krippendorff [11]. To supplement the alpha, we use two additional measures which are intended to capture agreement on the individuation of annotated behaviors.

Disagreement Type Analysis

As mentioned in the previous section, Time-Slice Krippendorff's alpha does not differentiate between disagreement caused by misalignment of the annotations or that caused by direct event disagreement. To better understand these annotation differences, we use two new metrics (see Figure 4), which can be valuable information when deciding whether crowd workers' training should concentrate on better behavior identification, segmentation, or both.

- *Event Agreement Metric.* An agreed event is defined as when there is an overlap of identified events in two annotations. In other words, agreed events are those that both annotators jointly identified. Depending on which annotation is taken as the reference point however, the number of agreed events could be different (see Figure 4). For this reason, we compute the percentage of agreed behavior events between the two annotations by dividing the total number of agreed events from both reference points by the total number of identified events from both reference points.
- *Segmentation Agreement Metric.* Another informative measure in gauging the agreement between two annotators is to see how precisely they segmented the boundary of the same annotation event. To compute the segmentation precision, we look at the time windows of agreed behavior events from both reference points combined and compute agreement within the time windows only (see Figure 4). The percentage is computed by dividing the number of agreed time slices by the number of total time slices within the time window of agreed events.

Datasets

From YouTube, which is a video-sharing website where users can upload and share videos, about 360 videos of people giving movie reviews were collected. Each video was annotated by two coders to determine the sentiment of the reviews (negative, neutral, and positive). From those videos, 20 videos were selected for this study that were both gender-balanced and sentiment-balanced (to have various expressions). Additionally, 5 more videos were randomly selected and used for training purposes. Each video showed a frontal, upper-body shot of a different person talking. Since all of the videos appeared to have been recorded using a webcam, it should be noted that the overall quality of the videos was not ideal but still fair enough to discern various facial expressions and eye gaze. For the 20 videos that were used in the actual experiments, the frame rate was at 30 frames per second and the video length ranged from 60 to 180 seconds, averaging at 138 seconds. The 5 training videos had the same frame rate, averaging at 106 seconds in length.

To show the generalization of our training procedure, a second dataset was created with 10 clipped videos from the Semaine corpus [16], which is a well-known video corpus in the research communities focusing on emotion, affective computing, and human behavior analysis. The purpose of this second dataset was to investigate if the effect of training crowd workers on one dataset can be transferred to another dataset for annotating human behaviors. These videos also showed a frontal, upper-body shot of a person speaking, and the frame rate was also at 30 frames per second, averaging at 150 seconds in length.

Annotated Behaviors

From behaviors that were relatively common and frequent in all the videos, we selected 4 different types of behaviors to annotate based on their variety (one for eyes, one for facial expressions, one for head movements, and one for verbal cues) and difficulty. These behaviors are all very frequently annotated behaviors for research involving human behavior analysis. The descriptions of the behaviors in our coding schemes were adapted from the MUMIN multimodal coding scheme [1].

- Gaze away: eye gaze is directed away from the camera.
- Pause filler: the person says “um...” or “uh...”
- Frown: the eyebrows contract and move toward the nose.
- Headshake: a repeated rotation of the head from one side to the other.

Experimental Design

We used Amazon Mechanical Turk (AMT) for our experiments, which is arguably the most well-known and widely used platform for crowdsourcing. The main idea behind AMT is to distribute small tasks at which humans are proficient and computers are still incompetent to a crowd of workers worldwide. Using AMT’s web interface, the “requesters” can design and publish tasks online, which are called HITs for Human Intelligence Tasks. In designing HITs, the requesters can set various options to restrict access to specific kinds of workers, set the number of unique workers to work on them, and set the amount of monetary reward. Moreover, a HIT template can be created, and one can define variables whose values will vary from HIT to HIT, which becomes very useful in creating a batch of similar HITs but with different videos. We created a HIT template with OCTAB annotation interface integrated and batch created all of our HITs with the videos in our YouTube and Semaine datasets. A total of 19 workers participated in our experiments, who worked for an effective hourly wage between $4 and $6 for compensation. The reader is referred to [15] for more detail on using AMT.

Experienced Local Coders

Two experienced local coders were recruited for this study, and the agreement between them after training was considered as the gold standard in our experiments. They devised coding schemes for the 4 behaviors to annotate, and they trained themselves to reach agreement on the 5 YouTube videos set aside for training purposes only. They trained on one video at a time until agreement (measured with Time-Slice Krippendorff’s alpha) reached a threshold of 0.80 (or very close) for all behaviors with the exception of headshake behavior because the local coders could not manage to reach 0.80 for some training videos even after 3 trials. However, the average alpha level for headshake behavior across the 5 training videos still reached 0.80. We performed a more detailed analysis of the types of errors in our experiments (see Results section) to better understand this challenge with headshake behavior (see Figure 7, bottom part).

After training, each of the local coders used the same environment as crowd workers to annotate all the videos from the YouTube and Semaine datasets across all the behaviors. Since agreement between the local coders was high, the final annotations from one of the local coders during training were used as ground-truth annotations to train crowd workers online.

Untrained Crowd Workers

To compare our approach with a scenario where crowd workers are untrained, we selected a total of 12 workers to participate as untrained crowd workers. As mentioned earlier, they were screened using an annotation test for gaze away behavior. This brief screening process was only to make sure that they could pay attention to frame-level detail, and no training sessions were given. They were provided with the coding schemes drafted by the two local coders, and they made a combined effort to annotate all the videos from only the YouTube dataset across all the behaviors.

Trained Crowd Workers

A total of 7 workers, who were not involved as untrained crowd workers, participated as trained crowd workers. They were trained with the same 5 YouTube videos that local coders used for training. After each training video, workers received e-mail feedback with our OCTAB training module, generated automatically with scripts. Workers were considered trained when they reached the same alpha thresholds used for experienced local coders. The training process involved only at most 1 trial per training video for gaze away and pause filler behaviors. For frown behavior, each worker took mostly 1 trial per video to reach the alpha threshold on average across all training videos, and it took about 2 to 3 trials per training video for headshake behavior. The trained workers were provided with the coding schemes drafted by the two local coders and annotated all the videos across all the behaviors from the YouTube dataset first. Then, they similarly annotated the Semaine dataset to investigate if the effect of training crowd workers for annotating human behaviors on one dataset can be transferred to annotating a different and independent dataset. The crowd workers were not informed that these videos were from a different dataset.

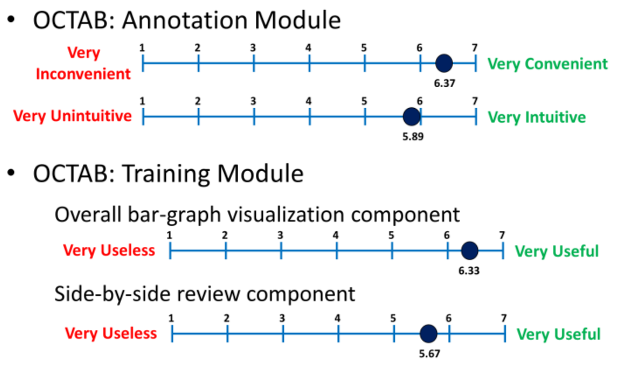


Figure 5. The user experience ratings of our OCTAB interface.

Repeated Annotations

For both of the above-mentioned conditions with untrained and trained crowd workers, 3 repeated annotations were obtained to investigate the benefit of taking a majority vote approach.

Annotation Strategies

For each dataset, we compared the agreement performance of three annotation approaches: *experts*, *crowdsourced unique*, and *crowdsourced majority*.

Experts. We had two local experienced coders who each produced a complete set of annotations for each dataset. The agreement between the two local coders was considered as the gold standard in our experiments. We refer to these sets as *experts* in the next section.

Crowdsourced Unique. From crowd workers, we obtained 3 repeated annotation sets from different workers per behavior per video. By randomly permuting the order in the 3 annotation sets, we created 3 complete sets of crowdsourced annotations for each dataset, which we refer to as *crowdsourced unique*.

Crowdsourced Majority. The 3 complete sets of crowdsourced annotations can be combined to make another complete set using majority voting, where a time slice (or frame) is judged annotated if at least 2 out of 3 workers agreed. We refer to this set as *crowdsourced majority* for each dataset.

We compared agreement in three different combinations: (1) within *experts* so that we have a baseline, (2) *experts* vs. *crowdsourced unique* to see if having one worker annotate per video is sufficient, and (3) *experts* vs. *crowdsourced majority* to see the benefit of having repeated annotations and performing a majority vote. The agreement comparison was performed for the YouTube dataset with untrained crowd workers, the YouTube dataset with trained crowd workers, and the Semaine dataset with trained crowd workers.

RESULTS AND DISCUSSIONS

This section highlights five main research problems studied during our experiments: the user experience ratings of

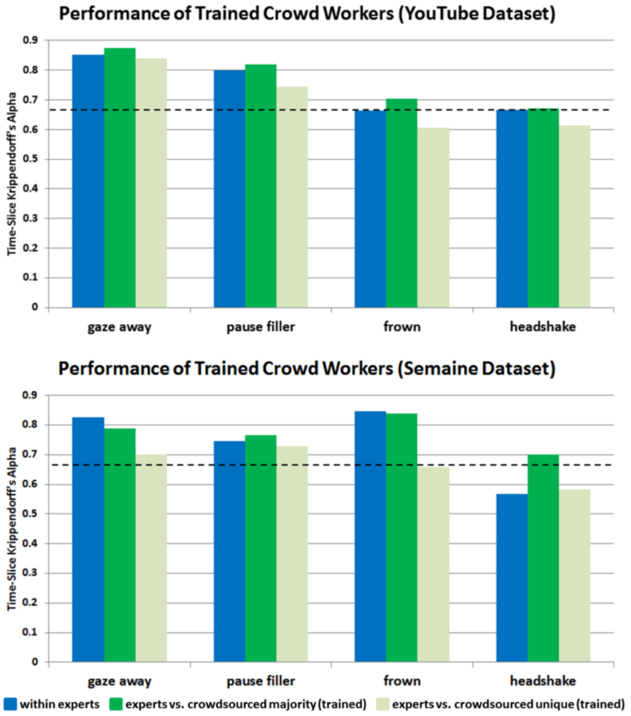


Figure 6. The performance of the trained crowd workers on the YouTube dataset (top) and the Semaine dataset (bottom). The dotted lines indicate the agreement alpha threshold point at 0.667.

OCTAB annotation and training modules, the performance of trained crowd workers, the performance of untrained crowd workers, the analysis of the types of disagreement, and the sensitivity analysis of Time-Slice Krippendorff's alpha measure to test its stability and reliability.

It should be noted that researchers in social sciences usually consider macro-level annotation data with a Krippendorff's alpha value equal to or above 0.80 as reliable and in high agreement, and they consider data with an alpha value equal to or above 0.667 but lower than 0.80 as reliable only to draw tentative conclusions [10]. These threshold points, however, are somewhat arbitrary, and it is controversial whether the same standards are fair to hold for judging the reliability and quality of micro-level (frame-level) behavior annotations. Keeping this in mind, we nevertheless use the 0.667 threshold as the standard of quality in the remainder of this section.

User Experience Ratings of OCTAB

The 19 crowd workers who participated in the experiments completed a survey to evaluate OCTAB annotation and training modules and the behavior annotation tasks (see Figure 5). On a 7-point Likert scale to rate OCTAB annotation module's convenience (from very inconvenient at 1 to very convenient at 7) and intuitiveness (from very unintuitive at 1 to very intuitive at 7), the mean score was 6.37 (n = 19, sd = 0.74) for convenience and 5.89 (n = 19, sd = 0.91) for intuitiveness. For OCTAB's training module,

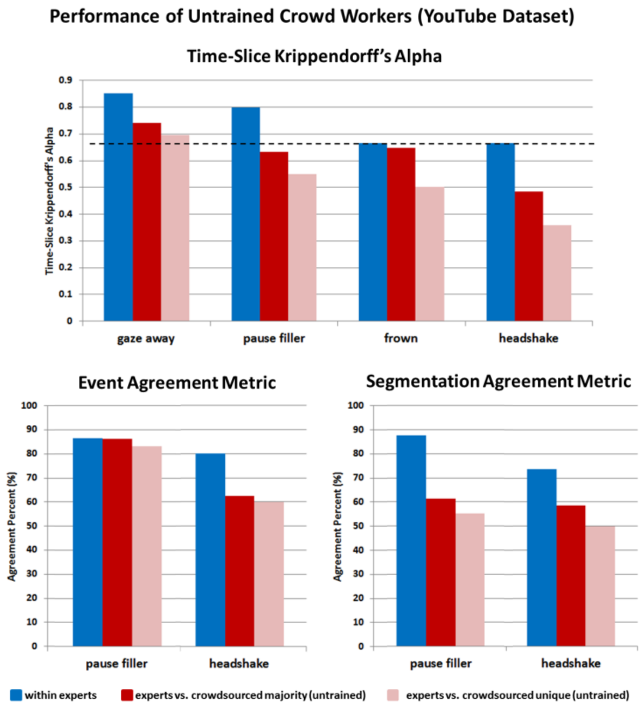


Figure 7. The performance of untrained crowd workers on the YouTube dataset. The dotted line indicates the agreement alpha threshold at 0.667.

the mean score on usefulness (from very useless at 1 and very useful at 7) was 6.33 (n = 6 sd = 0.47) for the bar graph visualization and 5.67 (n = 6, sd = 1.97) for the side-by-side review component. These evaluation results show high usability of our OCTAB interface.

The crowd workers also evaluated the difficulty of each behavior to annotate (from very difficult at 1 to very easy at 7), and the mean score was 6.42 (n = 19, sd = 0.82) for gaze away behavior, 5.71 (n = 14, sd = 1.33) for pause filler behavior, 3.94 (n = 16, sd = 2.05) for frown behavior, and 3.64 (n = 14, sd = 1.59) for headshake behavior. Not surprisingly, the reported difficulty level correlated with the general agreement performance of each behavior.

Performance of Trained Crowd Workers

For the YouTube dataset, on which the crowd workers were trained to perform the annotation tasks, the performance of *crowdsourced majority* was striking. For all behaviors, the average agreement between individual experienced local coders and *crowdsourced majority* was higher than between the two local coders themselves (see Figure 6). The average alpha between *experts* and *crowdsourced majority* reached above the 0.667 threshold for all behaviors, specifically 0.87 for gaze away behavior, 0.82 for pause filler behavior, 0.70 for frown behavior, and 0.67 for headshake behavior. These results show that crowdsourcing can be a very effective tool for researchers in obtaining high-quality behavior annotations, provided that proper training sessions were given and 3 repeated annotations were obtained to take a majority vote approach. For relatively unambiguous behaviors, such as gaze away and pause filler behaviors, the result indicates that repeated annotations are actually unnecessary and having one worker annotate per video is sufficient to obtain high-quality annotations.

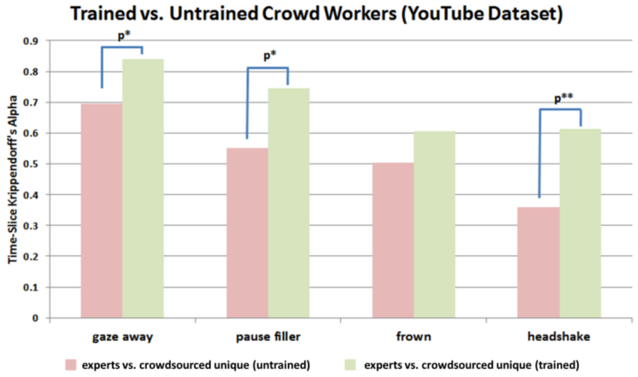


Figure 8. The performance comparison between the untrained and trained crowd workers on the YouTube dataset (t-tests showed statistically significant difference at p* < 0.01 and p < 0.001).**

When the crowd workers, who were trained on the YouTube dataset, performed the same annotation tasks on different videos in the Semaine dataset, we could observe the effect of training actually transferrable. The agreement between *experts* and *crowdsourced majority* was almost equal to or higher than between *experts* themselves except for gaze away behavior. This exception is most likely due to the speakers in the Semaine videos not talking directly toward the camera as was the case in YouTube dataset. The speakers in the Semaine dataset talk to an interlocutor (invisible in videos) and this difference probably introduced much confusion in deciding what makes a gaze away behavior in the changed setting because the coding scheme was the same for both datasets. Nevertheless, the average alpha between *experts* and *crowdsourced majority* was still high at 0.79 for gaze away, 0.77 for pause filler, 0.84 for frown, and 0.70 for headshake. We can also observe a similar trend that having only one worker annotate per video is sufficient to obtain high-quality annotations for gaze away and pause filler behaviors.

Performance of Untrained Crowd Workers

The performance of the untrained crowd workers on the YouTube dataset shows that both *crowdsourced unique* and *crowdsourced majority* reached the agreement alpha threshold of 0.667 for gaze away behavior (see Figure 7). The agreement between *experts* and *crowdsourced majority* reached very close to the 0.667 threshold for pause filler and frown behaviors, and it should be noted that it is not uncommon for an alpha value of 0.60 to have well over 85% of agreement at the frame-level without chance correction, which is by no means a low agreement.

The result also shows the benefit of disagreement analysis with event and segmentation agreement metrics. For instance, the disagreement analysis reveals that the source

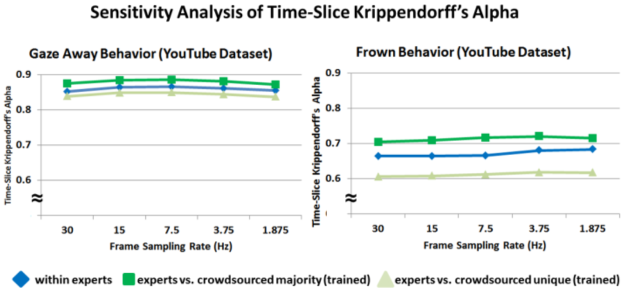


Figure 9. Sensitivity analysis of Time-Slice Krippendorff's alpha across different frame sampling rate.

of the low alpha value for crowd workers in annotating pause filler behavior was not in behavior identification but in segmentation. In other words, the untrained crowd workers were just as proficient as the experienced local coders in identifying instances of pause filler behavior. However, for headshake behavior, the disagreement analysis shows that the untrained crowd workers had problems of both identifying and segmenting behaviors correctly compared to the experienced local coders. This analysis is aligned with our previous observation that experienced local coders also had trouble agreeing with an alpha threshold at 0.80.

Trained vs. Untrained Crowd Workers

We emphasize the effect of training in Figure 8, which shows the average agreement alpha values between *experts* and trained *crowdsourced unique* and also between *experts* and untrained *crowdsourced unique*. By training crowd workers, their agreement performance on the YouTube dataset improved with a statistical significance at $p < 0.01$ for gaze away and pause filler behaviors and at $p < 0.001$ for headshake behavior (statistical significance computed with t-tests).

Time-Slice Krippendorff's Alpha

For all behaviors, Time-Slice Krippendorff's alpha was shown to be a stable measure that stayed consistent across different sizes of time slices, and we show the results for gaze away and frown behaviors on the YouTube dataset in Figure 9. For this experiment, annotation sets created at lower frame rate were up-sampled using a majority vote technique, where each time slice was considered annotated if at least 50% of the slice was annotated.

CONCLUSIONS

This paper presents a novel web interface and training procedure for crowdsourcing micro-level behavior annotations in videos and shows that such annotations can achieve a quality comparable to those done by experienced local coders. Specifically, we presented an effective web tool called OCTAB for crowdsourcing micro-level behavior annotations online, which consists of a convenient and precise annotation module and a training module that give crowd workers the ability to quickly get trained by seeing first an overall view of their errors and then performing side-by-side review of their annotations against ground-truth annotations. Our results from an extensive set of experiments showed the feasibility of our crowdsourcing approach for obtaining micro-level behavior annotations in videos, showing the reliability improvement in annotation accuracy when properly training online crowd workers. We also investigated the generalization of our training approach to a new video corpus, showing that our training procedure is transferrable across different independent video corpora.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. IIS-1118018 and the U.S. Army Research, Development, and Engineering Command (RDECOM). The content does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

We thank the USC Annenberg Graduate Fellowship Program for supporting the first author's graduate studies.

REFERENCES

1. Allwood, J., Cerrato, L., Dybkjaer, L., Jokinen, K., Navarretta, C., and Paggio, P. The MUMIN multimodal coding scheme. *Proc. Workshop on Multimodal Corpora and Annotation 2004.*
2. Bernstein, M., Brandt, J., Miller, R., and Karger, D. Crowds in two seconds: Enabling realtime crowd-powered interfaces. *Proc. UIST 2011*, 33-42.
3. Biel, J. I. and Gatica-Perez, D. The good, the bad, and the angry: Analyzing crowdsourced impressions of vloggers. *Proc. ICWSM 2012*, 407-410.
4. Dasiopoulou, S., Giannakidou, E., Litos, G., Malasioti, P., and Kompatsiaris, Y. A survey of semantic image and video annotation tools. *Knowledge-Driven Multimedia Information Extraction and Ontology Evolution*, *Lecture Notes in Computer Science 6050* (2011)*,* 196-239.
5. Downs, J. S., Holbrook, M. B., Sheng, S., and Cranor, L. F. Are your participants gaming the system?: screening mechanical turk workers. *Proc. CHI 2010*, 2399-2402.
6. Gao, Q. and Vogel, S. Consensus versus expertise: A case study of word alignment with Mechanical Turk. *Proc. CSLDAMT 2010*, 30-34.
7. Hsueh, P. Y., Melville, P., and Sindhwani, V. Data quality from crowdsourcing: A study of annotation selection criteria. *Proc. ALLNP 2009*, 27-35.
8. Kang, S., Gratch, J., Sidner, C., Artstein, R., Huang, L., Morency, L. Towards building a virtual counselor: Modeling nonverbal behavior during intimate self-disclosure. *Proc. AAMAS 2012*, 63-70.
9. Kim, J., Nguyen, P., Weir, S., Guo, P., Miller, R., and Gajos, K. Crowdsourcing step-by-step information

extraction to enhance existing how-to videos. *Proc. CHI 2014*.

10. Krippendorff, K. *Content Analysis: An Introduction to Its Methodology*. Sage, Beverly Hills, CA, 2004.

11. Krippendorff, K. On the reliability of unitizing contiguous data. *Sociological Methodology* 25 (1995), 47-76.

12. Le, J., Edmonds, A., Hester, V., and Biewald, L. Ensuring quality in crowdsourced search relevance evaluation: The effects of training question distribution. *Proc. SIGIR CSE 2010*, 21-26.

13. Lew, M. S., Sebe, N., Djeraba, C., and Jain, R. Content-based multimedia information retrieval: State of the art and challenges. *ACM T. Multim. Comput. 2*, 1 (2006), 1-19.

14. Marge, M., Banerjee, S., and Rudnicky, A. I. Using the Amazon Mechanical Turk for transcription of spoken language. *Proc. ICASSP 2010*, 5270-5273.

15. Mason, W. and Suri, S. Conducting behavioral research on Amazon’s Mechanical Turk. *Behavior Research Methods 44*, 1 (2012), 1-23.

16. McKeown, G., Valstar, M.F., Cowie, R., and Pantic, M. The SEMAINE corpus of emotionally coloured character interactions. *Proc. ICME 2010*, 1079-1084.

17. Novotney, S. and Callison-Burch, C. Cheap, fast and good enough: Automatic speech recognition with non-expert transcription. *Proc. HLT 2010*, 207-215.

18. Nowak, S. and Ruger, S. How reliable are annotations via crowdsourcing: A study about inter-annotator agreement for multi-label image annotation. *Proc. MIR 2010*, 557-566.

19. Pantic, M., Pentland, A., Nijholt, A., and Huang, T. Human computing and machine understanding of human behavior: A survey. *Proc. ICMI 2006*, 239-248.

20. Park, S., Mohammadi, G., Artstein, R., and Morency, L.-P. Crowdsourcing micro-level multimedia annotations: The challenge of evaluation and interface. *Proc. CrowdMM 2012*, 29-34.

21. Poppe, R. A survey on vision-based human action recognition. *Image Vision Comput. 28*, 6 (2010), 976-990.

22. Quinn, A. J. and Bederson, B. B. Human computation: A survey and taxonomy of a growing field. *Proc. CHI 2011*, 1403-1412.

23. Rashtchian, C., Young, P., Hodosh, M., and Hockenmaier, J. Collecting image annotations using Amazon's Mechanical Turk. *Proc. CSLDAMT 2010*, 139-147.

24. Riek, L., O’Connor, M., and Robinson, P. Guess what? A game for affective annotation of video using crowd sourcing. *Proc. ACII 2011*, 277-285.

25. Ross, J., Irani, L., Silberman, M., Zaldivar, A., and Tomlinson, B. Who are the crowdworkers?: Shifting demographics in Mechanical Turk. *Proc. Ext. Abstracts CHI 2010*, 2863-2872.

26. Sheng, V. S., Provost, F., and Ipeirotis, P. G. Get another label? Improving data quality and data mining using multiple, noisy labelers. *Proc. KDD 2008*, 614-622.

27. Snow, R., O'Connor, B., Jurafsky, D. and Ng, A. Y. Cheap and fast but is it good?: Evaluating non-expert annotations for natural language tasks. *Proc. EMNLP 2008*, 254-263.

28. Spiro, I., Taylor, G., Williams, G., and Bregler, C. Hands by hand: Crowd-sourced motion tracking for gesture annotation. *Proc. CVPRW 2010*, 17-24.

29. Vondrick, C., Ramanan, D., and Patterson, D. Efficiently scaling up video annotation with crowdsourced marketplaces. *Computer Vision - ECCV 2010 6314*, 610-623.

30. Wu, S. Y., Thawonmas, R., and Chen, K. T. Video summarization via crowdsourcing. *Proc. Ext. Abstracts CHI 2011*, 1531-1536.

31. Yuen, M. C., King, I., and Leung, K. S. A survey of crowdsourcing systems. *Proc. SocialComp 2011*, 766-773.

32. Zhang, H., Law, E., Miller, R., Gajos, K., Parkes, D., and Horvitz, E. Human computation tasks with global constraints. *Proc. CHI 2012*, 217-226.