CrowdNote: Turning Wisdom and Effort of Crowds into Complex Media Annotation

Marcello N. de Amorim
Federal University of Espírito Santo
novaes@inf.ufes.br

Fábio R. de A. Neto
Federal University of Espírito Santo
fabio.ribeiro.neto@gmail.com

Celso A. S. Santos
Federal University of Espírito Santo
saibel@inf.ufes.br

ABSTRACT

This paper presents a method to achieve complex video annotation without requiring improved annotation tools, experts nor trained workers. In this method, the complex annotation process is divided into a set of simple annotation microtasks, and based on them is defined a workflow for generating complex annotation. Each of these microtasks is treated as a human computation function, producing an output that can be used as input to the next microtask in the workflow. In this way, the complex annotation production workflow is treated as a human computation algorithm. To demonstrate the operation of the method was developed a video enrichment system and was carried out an experiment in which the crowd was responsible for: 1. identify the points of interest; 2. suggest extra content; 3. Select the best content for each point of interest; and 4. position them in the scenes. This system was built using the framework developed to support the method and that can handle contributions from internal groups, public groups, and platforms such as Amazon Mechanical Turk, Crowdflower, and Microworkers.

CCS CONCEPTS

• **Information systems → Multimedia information systems**; **Crowdsourcing**; • **Human-centered computing → Web-based interaction**; **Computer supported cooperative work**; • **Applied computing → Annotation**;

KEYWORDS

Crowdsourcing, Media Annotation, Video Annotation, Human Computation, Microtasks, Multimedia Systems, Video Enrichment

ACM Reference Format:
Marcello N. de Amorim, Fábio R. de A. Neto, and Celso A. S. Santos. 2017. CrowdNote: Turning Wisdom and Effort of Crowds into Complex Media Annotation. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Media annotation consists of supplementing media objects, such as videos, images and audios, by adding metadata about their content and context, also to describing media characteristics such as quality, encoding, among other features [51]. This supplementary information can be used to make easier the work of users and systems that can handle annotated items [6]. It allows highlighting key points as well as add information to content presented[11], facilitating the creation of media applications for content-based distribution [54], indexing [55], summarization [16], navigation [20], composition [52], among many others, by both automatic and manual means [32, 50].

In this paper, media annotations are categorized as simple and complex ones, considering that simple annotations are those that can be acquired with a simple interaction of the workers in a microtask. Complementarily, a complex annotation is one that requires the worker execute a more tedious, hard or time-consuming task, in which he needs to perform multiple interactions.

Automatic methods for media annotations often present satisfactory efficiency and interesting results, though, they generally apply techniques that require well-structured media objects and extensive examples database, such as deep learning[31]. Unfortunately, many scenarios cannot provide these requirements, making it impossible to use automatic methods for video annotation [35].

In another way, manual media annotation is suitable for these scenarios because it uses human intelligence to handle the tasks. However, manual video annotation can be high-costly because of the potentially high-density of annotation points in the video, as well as the complex nature of some annotation tasks.

An alternative to achieving a media annotation in a general scenario is to employ collaborative or cooperative approaches, which are differentiated in this paper. In a collaborative approach, the contributors work together to solve the main problem. Otherwise, in a cooperative approach, each contributor solves a part of the main problem to produce a final result [33].

Taking cooperative approaches to a higher level, crowdsourcing media annotation has emerged as a proposal to annotate media objects using a large number of contributors efficiently [47]. Following the crowdsourcing principles, the tasks distributed to the workers are modeled to be done independently, maximizing the parallelism [26]. Moreover, each task can be sent to many contributors, making possible to compare, check and to aggregate the contributions also reducing the chance of producing a biased result [19].

A frequent problem of using a crowdsourcing approach to media annotation is to balance the relationship between task complexity and cost. Simple annotation tasks, such as clicking an object on a video, can be done in a few seconds for anyone. Otherwise, more

complex tasks such as providing complementary content and positioning it in the right position on a video, require some expertise of contributors and are more costly to them. In a crowdsourcing context, microtask is a ubiquitous designation for simple tasks that can be performed for any contributor quick and easily [15].

The method presented in this paper aims to achieve a complex media annotation without requiring trained workers or experts, employing a set of simple annotation tools rather than complex and expensive annotation systems. In this way, a complex annotation process is divided into a set of simple annotation microtasks, and based on them is defined a workflow to generating the outcome. This aims to provide ways to get around some problems faced in achieving a complex media annotation:

- By using manual annotations, no example bases or restricted conditions are required as in automatic methods.
- By using a microtask-based crowdsourcing process is not required experts nor trained workers. Also, it makes the contribution process simple and quick, avoiding time-consuming and tedious tasks to workers.
- By using microtasks in which only a simple annotation is collected is not required sophisticated annotation tools.

Following this method, each simple annotation microtask is modeled here as a process composed by two step: Collection and Aggregation. In the Collection step the contributions are received from the crowd, and in the Aggregation step these contributions are processed in order to generate its output.

Also, each microtask is treated as a human computation function, producing an output that can be used as input to the next one in the workflow. In this way, the complex annotation production workflow is treated as a human computation algorithm. This point of view allows to design some features such as generate multiply outputs from a microtask to create simple outcomes from each partial result. These outcomes can be datasets, sumaries, marks and more, so a complex annotation process can generate multiple output artifacts.

A previous version of the method, called CrowdNote, was introduced with some limitation [6]. This updated version of CrowdNote has incorporated the human computation algorithm concept, which allows to create more flexible production workflows and to produce multiple outputs for each microtask. The scope of the method has also been expanded, including annotations for various media types instead of just videos.

This version also supports automatic, manual and supervised aggregation methods, instead, only automatic ones. Supervised methods are specially useful when is not possible specify rules or algorithms to generate an output from the contributions, this usually involves subjectivity, emotions and other human abilities.

To demonstrate the operation of the method was developed a video enrichment system that was built over a flexible architecture that can handle contributions from internal groups, public groups, and platforms such as Amazon Mechanical Turk, Crowdflower, and Microworkers. Then, an experiment was conducted in which the crowd was responsible for:

- Identify the points of interest.
- Suggest extra content.
- Select the best content for each point of interest.
- Position them in the scenes.

The rest of this paper is structured as follows. Section 2 presents the concepts used and how they are employed in the proposed approach. Section 3 presents related works. Section 4 presents the extended CrowdNote method and framework. Section 5 presents the conducted experiment. Finally, section 5 concludes the paper presenting final considerations and future prospects.

2 RELATED CONCEPTS

The presented method is based on the human computation paradigm [47], using a crowdsourcing approach [26] to annotating media objects. Its goal is to generate complex annotations without the need for experts nor sophisticated annotation tools. In this method, complex annotations are generated by a composition of simple annotation tasks, which must be modeled so that they can be performed by individuals from a crowd of workers. To do so, this method employs some fundamental concepts commonly associated with human computation. These concepts are presented in this section.

2.1 Human Computation

There are jobs that are trivial for all people, even for kids, but they are extremely difficult even for the most powerful and sophisticated software and software. This type of task has as its characteristics the need for creativity and level of abstraction which, in the present moment, is slightly in the minds of human beings.

Some examples classify this type of task with image recognition, especially when there is occlusion or involves subjective analysis, content authorship, analysis of emotions and so many other activities that inherently require human intelligence to be performed. Luis von Ahn introduced in his dissertation [47] a paradigm named Human Computation that allows to approach the problems from this point of view, identifying in it what tasks can be automated and which are those that require human treatment. Additionaly Human Computation can improve performance by division of labor because it helps to define tasks that can be executed in parallel [40].

One of the benefits of modeling a system according to the Human Computation paradigm is to focus the effort of human collaborators only on tasks that really require their attention, this is done by identifying the tasks that inherently require human intelligence. Each of these tasks is a HIT (Human Intelligence Task) and corresponds to something that humans can easily solve while a machine presents extreme difficulty in trying to solve [30].

In short, Human Computation is a paradigm that proposes to identify into the problems which tasks require human intelligence and which tasks can be automated. In general, it may be benefited with modeling based on the Human Computation paradigm the problems in which it is possible to identify tasks that are very difficult for machines but which can be easily completed by humans. In terms complexity, a HIT can be a microtask or a macrotask. These strategy is ilustrated in 1.

Macrotasks require more effort from the worker, often requiring him to be an expert in the field or to have training on subjects related to the task. In the context of media annotation, this type of task is suitable for complex annotations because it assumes that the worker is qualified and will devote the time and effort required to complete it. However, macrotasks often require sophisticated

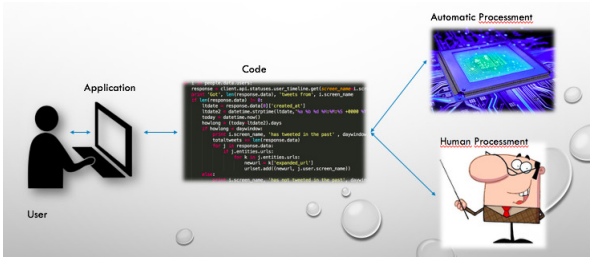

Figure 1: Human Computation

annotation systems and limit the group of workers who are able to execute them [22]. In this work, complex annotations are defined as those that combine annotations on different aspects of annotated objects and therefore are usually obtained by macrotasks.

On the other hand, microtasks are usually modeled in a way that can be accomplished quickly and easily by less skilled workers. For media annotation, this kind of task is usually used to generate simple annotations, which refer to one or a few items to be annotated in each task. Also, often a Microtask can be performed using a simple annotation tool.

Microtasks are widely used in crowdsourcing projects and this kind of task is supported by well-established commercial platforms such as Amazon Mechanical Turk, CrowdFlower and Microworkers. According to these requirements microtasks should be:

- **Small:** a worker must complete a task by means of few interactions, preferably by a single interaction.
- **Quick:** it should be possible to complete a task in a very short time, preferably within a few minutes.
- **Easy:** the easier the task, the less skilled the workers should be. Preferably, a task must be modeled so that it can be performed by any worker, just read the instructions and have the technical requirements for the task, such as minimal screen resolution, audio devices, or minimal Internet connection speed.

Moreover, considering each microtask as a human computing function, mapping elements from input to output, it is possible to understand that it is possible to compose an algorithm using these functions to obtain equivalent results generated by macrotasks [9].

2.2 Crowdsourcing

Human computation approaches can improve performance by division of labor because it helps to tasks that can be executed in parallel. Each worker performs their work independently, so that the instances of a task can be executed in parallel, according to the Human Computation paradigm [40].

To support this kind of cooperative process, crowdsourcing has emerged as a proposal to annotate media objects using a large number of contributors efficiently [47]. Following the crowdsourcing principles, the tasks distributed to the workers are modeled to be done independently, maximizing the parallelism [26]. Moreover, each task can be sent to many contributors, making possible to compare, check and to aggregate the contributions also reducing the chance of producing a biased result [19].

This approach generally delivers good quality results using contributions from a crowd of contributors and can distribute, collect, validate and combine large amounts of contributions [22, 24, 34]. Since this approach is designed to handle a huge number of collaborators and contributions for tasks that require human intelligence [26], Crowdsourcing is appropriate to allow the Human Computing paradigm to be applied in a massive-scale online collaboration [4].

Crowdsourcing is supported by four pillars: The Crowdsourced Task, The Crowdsourcer, The Crowd, and The Crowdsourcing Plataform [25].

- **The Crowdsourced Task** is the HIT designed, according to the Human Computing paradigm, to acquire workers' contributions. Instances are generated of the task that are presented to the workers as jobs that must be performed [15].
- **The Crowdsourcer** is the owner of a project, it may be an individual or institution that wishes to have a completed task. The owner is responsible for starting the crowdsourcing process, defining what task must be completed and how it should be presented to the workers as jobs [25].
- **The Crowd** is the work force that moves the process once it is composed of all the workers who perform the jobs needed to generate the outcome. Each worker carries out his work independently, so that the works can be executed in parallel, according to the paradigm of Human Computation [40].
- **The Crowdsourcing Platform** is the centralizer of the whole proces, serves as an entry point for both the owner making the tasks available and for the workers to execute them. This kind of environment can be something sophisticated such as CrowdFlower, Microworkers, and AmazonMechanical Turk [15], or really simple systems with screens and forms for data collection such the mobile application used by the Google Crowdsource project [2]. A crowdsourcing environment is necessary, as the tasks must be made available to a potentially large number of workers. The crowdsourcing platform is a key element of support to massive-scale collaboration.

The use of a commercial crowdsourcing platform brings benefits such as not having to worry about management's issues of workers, jobs and contributions such as employee recruitment and collection of contributions as well as facilitating employee payments. Also, payouts are a good way to motivate and keep the crowd, although there are other coping factors such as personal accomplishment and even methods of gambling.

To exemplify how Crowdsourcing adds support for massive-scale online collaborations to human computer systems, one can analyze the collaborative processes involved in Luis von Ahn's three most well-known projects, reCAPTCHA [44], ESP Game [39], and Duolingo [48].

ESP Game used a game as an input interface, so players who score points by adding tags that describe the images presented to them are viewed by the platform as workers annotating a base of images [39]. CAPTCHA tests are very popular on the internet, being widely used by various applications and sites, reCAPTCHA collects the responses that users provide for these tests, and so the platform also sees these interactions as workers annotating images

reCAPTCHA [44]. In the Duolingo users carry out translations while learning and practicing another language, and the platform collects these translations as contributions from the workers [3].

It is possible to observe in these examples, that in addition the modeling according to the Human Computation paradigm, present a viable way of distributing jobs and collecting results for a potentially huge crowd of workers so that these contributions can be used to complete a task.

2.3 The Wisdom of Crowds

Francis Galton wrote in 1907 an article in which he reported an experiment that a crowd of people at an agricultural fair tried to guess the weight of a particular ox. Galton verified that the average of the assumed weights converged to a value very close to the actual weight of the ox and analyzing the distribution of values grounded the Wisdom of the Crowd concept, according to which a heterogeneous crowd large enough tends to provide such a good expert result [19]. Additionally, James Surowiecki listed in his book "The Wisdom of the Crowds" [46] four requirements for a crowd to deliver good results.

- **Diversity:** each person adds private information or bias.
- **Independence:** people form their opinions independently.
- **Decentralization:** people draw on their own knowledge.
- **Aggregation:** a mechanism exists to turn private judgments into a collective decision.

Crowdsourcing platforms are environments suited to this theory, especially because it is possible to reach workers from different parts of the world and contexts with different backgrounds, which favors diversity. A common scenario in these environments involves workers receiving tasks and executing them without interacting with others, which helps create conditions for Independence and Decentralization.

For each task class, an appropriate aggregation method must be applied. These methods may involve convergence of responses, geometric means, contribution merging, as well as other types of processing. In this work, three categories of aggregation methods are also considered: automatic, supervised and manual.

- **Automatic:** an automatic aggregation method employs algorithms to process the contributions in order to generate the desired outcome. This kind of method involves convergence analysis, geometric mean, instance prevalence and ranking.
- **Supervised:** in some cases its possible to apply automatic convergence methods, although is required additional human verification or some human interaction in the contributions' processing.
- **Manual:** manual aggregation is used when its not possible to define the rules or algorithm that should be followed to generate the outcome. These situations are usually related to subjectivity and emotional aspects that must be analyzed in the contributions to generate the result.

3 RELATED WORK

Crowdsourcing video annotation approaches are used in various applications and are used to gather information of various types, such as temporal synchronization[12, 53], events[29, 45], scene objects[1, 37], emotions[8, 41], actions[13, 38], quality[17, 23], geotagging[10, 21], social relevance[7, 27, 43] and captions[14, 28].

However, some of these works are based on complex annotation tools, demanding hard, tedious or time-consuming tasks, or requiring trained and skilled workers. Some relevant examples that should be regarded include works such as [1, 8, 13, 21, 27, 36, 49].

VidWiki[1] is a complex system to improve video lesson by video annotation, which provides a complex annotation tool(Figure 2) that allows the worker to edit video scenes by adding various types of annotations, including LaTex equations. Another interesting paper to note was written in 2012 by C.Vandrick[49], in which time-consuming complex tasks were deployed in the Amazon Mechanical Turk[21] demanding specialized work to perform them.

While these works often produce interesting results, to adopt complex annotation tools, as well as hard and time-consuming tasks, restrict potential workers and owners capable of developing complex tools and hiring skilled workers.

Figure 2: VidWiki annotation tool[1]

There are also papers on crowdsourcing video annotation that report the use of simple tools and microtasks that can be done quickly by unskilled workers. These works include [9, 12, 18, 29, 37, 38, 45, 53].

The work published by N.Gagil in 2014[18] uses a very simple annotation tool(Figure 3) that allows the workers to perform an easy microtask, which consists of annotating videos with surveillance problems if any of them are found.

ReTool[9] is a work that must be mentioned because it presents a web-based tool for owners to create and publish annotation microtasks and workflows to execute them.

ToolScape[29] is a work that deserves prominence, as it is strongly related to the approach presented in this paper. ToolScape integrates simple annotation tools in which workers can perform a sequence of three microtasks, that was used to extract the step-by-step structure of the instruction videos, one of these tools is shown in Figure 4. Moreover, is presented a design pattern to define the workflow for these tasks.

This brief discussion about some related works aims mainly to highlight the characteristics of the microtasks, as well as the simple annotation tools used to execute them.

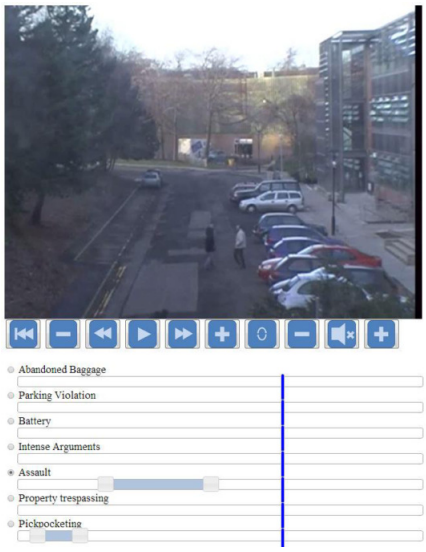

Figure 3: Simple surveillance annotation tool[18]

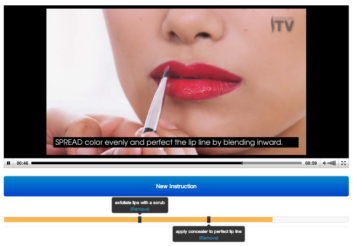

Figure 4: ToolScape annotation tool[29]

4 CROWDNOTE

4.1 Method

– Full Paper Webmedia 2017 [6]

The crowdsourcing video annotation approach presented in this paper follows three steps: Preparation, Annotation, and Presentation.

The preparation step describes how a complex annotation task can be divided into simple microtasks, in addition, is presented a workflow for the activities required before the annotation step, such as to define what should be annotated and the annotation types, as well as to design the microtasks and the simple annotation tools to execute them. In the annotation step, the annotation microtasks are performed by crowd workers, that are the contributors to the process. This step follows a workflow in which each microtask is followed by a specific aggregation method that generates a result so that the output from a task feeds the next one. The presentation step displays the outcome delivered by the annotation step, also at this point, all partial results are available to be used in other applications. The approach introduced also allows the development of expansive video annotation systems in which it is easily possible to add new microtasks to improve its result or generate new results.

These steps contain specific activities and are executed sequentially how can be seen in Figure 5.

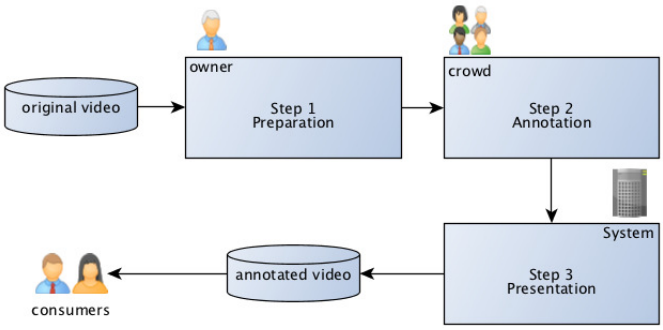

Figure 5: Process workflow

Preparation: all activities involved in this step are performed by the owner, who started the video annotation process. At this step is determined what must be annotated, also how they should be annotated. In this way the owner must determine:

(1) What kinds of point of interest should be annotated.
Ex: events, objects, subjects, issues.
(2) What annotation type will be used for each of these kinds.
Ex: free write, item selection, button click, image upload.
(3) What data type will be collected for each annotation type.
Ex: plain text, location, image, video.

To illustrate this point, the example of the football(soccer) match annotation will be recalled. In a football(soccer) match video the kinds of point of interest correspond to events such as goals, cards, and faults. For each point of interest observed it should be collected its kind, and the instant when this event happened. The annotation type to be used on the annotation tool can be a set of icons related to each event. Finally, the data type collected in this case may be plain text that contains the kind of event identified and the instant it happened [42].

Also, it is important to provide explanations or guidelines that can instruct the workers about how to execute the microtasks. An additional activity on the Preparation step is to determine what section of the video should be sent by each worker, this division can be made by duration (ex: send a 5 seconds segment to each worker), or using contextual criteria such as to send to each user a segment that contains a single dialog. The activities sequence for this step can be observed in Figure 6.

Annotation: An essential aspect of this step is to determine the microtasks' workflow, so the output from a task is taken as input by the next one, generating an outcome at the end of the last microtask. This cascade workflow is illustrated in Figure 11. It is important to notice that each task cell in composed by two activities, the microtask in self and the aggregation method, that generates the output from the obtained contributions. In this way, the output from the last task cell is the outcome provided by the system.

Presentation: at this step is generated an annotated video including the original video and the final outcome from the previous

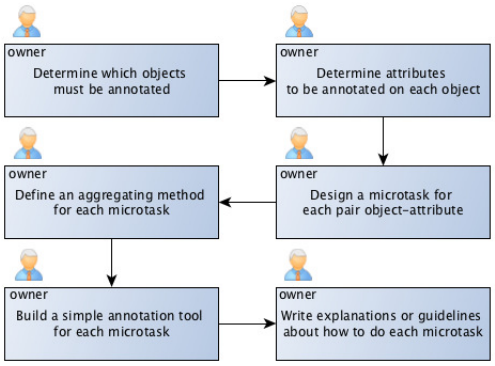

Figure 6: Preparation step

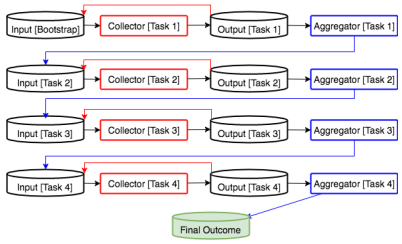

Figure 7: Annotation step for N microtasks

step. Other activities that can be proceeded at this step is to generate or to render, media items selected from the crowd annotations, as well as aggregate these items over the videos to compose a multimedia presentation.

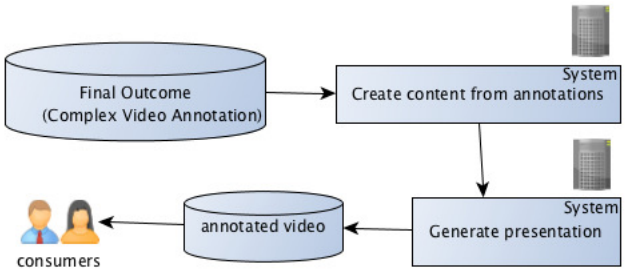

Figure 8: Presentation step

4.2 Framework

– WFA Webmedia 2017 [5]

CrowdNote was developed as a classic Web system. To facilitate the sharing of all produced software, only technologies that do not require complex infrastructure were adopted. The Server was fully developed in NodeJS for easy deployment, the Client was developed in HTML 5 to improve compatibility, and the Database uses MongoDB as No-SQL database for flexible persistence.

The architecture of the CrowdNote is illustrated in Figure 9 in which is possible to observe the 3 main components: Server, Database, and Clients.

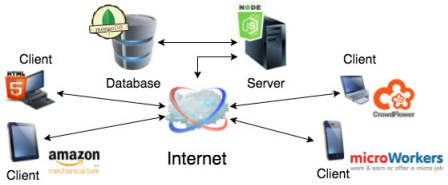

Figure 9: CrowdNote Architecture

4.3 The Server Component

The server system, illustrated in Figure 10, is composed of 3 modules: Collector, Aggregator and Player Provider.

- **Collector:** The Collector sends the jobs to the workers, receives the annotations from them, and stores the annotations into the Database. Information is exchanged between the Collector and the Client as JSON messages through HTTP requests for cross-platform compatibility.
- **Aggregator:** The Aggregator verifies, filters, groups, and processes the collected annotations of the crowd according to the rules defined for each task, and then stores the result in the Database.
- **Player Provider:** The Player Provider sends to the client the annotations, the extra content, and the original video. Thus, the player on the client can play the enriched video synchronously.

4.4 The Database Component

The persistence was addressed using MongoDB, which delivers a very attractive solution to build No-SQL databases with some characteristics that meet the crowdsourcing requirements such as high write load, high availability in an unreliable environment, easy scaling and partition, heterogeneous data into the same collection.

In this model, JSON document collections are used instead of tables, and the documents in each collection may have a different structure to store different attributes. This feature allowed the modeling of a very simple database structure, composed of 3 collections of documents. It was possible because documents in the Input and Output collections can contain different fields according to the task that consumes or generates the entries.

The Video collection stores entries related to the video segments dataset, the Input collection stores the input entries to the tasks, and the Output collection stores the contributions collected from the crowd. The result of the aggregation for each task is stored in the Input collection to be used by the next task, supporting the cascading tasks approach.

4.5 The Client Component

The client consists of simple forms-based annotation tools and a player capable of playing video and extra content synchronously. The client has been fully developed in HTML5, in the simplest way possible. For each task, a simple annotation tool was created to collect contributions.

The Client communicates with the Server through JSON messages and HTTP requests so that they can be deployed on different

systems and sites or even on crowdsourcing platforms such as Amazon Mechanical Turk, Crowdflower and Microworkers [15], as long as the JSON structure is respected. By using these platforms, the search and reward of workers are delegated to them, however, there is a financial cost involved in doing so.

4.6 Workflow

The 3 main components of CrowdNote communicate through data flows from **A** to **G**, as can be seen in the workflow in Figure 10.

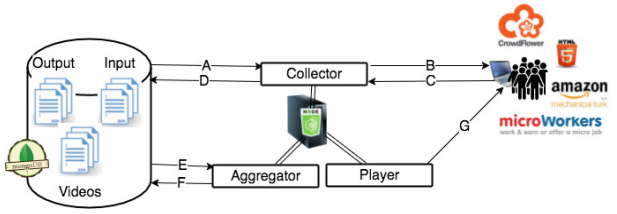

Figure 10: CrowdNote Workflow

- **A:** To generate each job to be sent to a worker, the Collector receives an entry from the Input Collection and the corresponding entry from the Videos Collection.
- **B:** The Collector sends a job to an instance of the Client, to be executed by a worker.
- **C:** The Client sends to the Collector the annotation made by the worker for the job received.
- **D:** The Collector stores in the Output collection the annotation collected from a worker.
- **E:** The Aggregator receives from the Output collection all annotations collected for a given task.
- **F:** The Aggregator stores the resulting entries from the aggregation process in the Input collection so that they are supplied as input to the next task.
- **G:** The outcome of the cascade of microtasks is sent by the Player Provider to the client so that it can play the video synchronously with the extra content.

5 CASE STUDY

CrowdNote is an environment that provides a collection of annotation tools, aggregation methods, and persistence models that can be selected, sequenced, and modified to generate different types of crowdsourcing applications based on video annotations. In order to create a system based on this environment, it is necessary to define the required microtasks sequence, and then select and specialize the resources provided by CrowdNote.

To demonstrate its working was built an instance of CrowdNote which consists of a system for video enrichment by adding extra content provided by the crowd. In this system, contributors are responsible for identifying the points of interest in the video, suggesting what content should be associated with each one, deciding the best suggestion for each point of interest, and finally deciding the best position in the video to present each content.

The extra content suggested by the crowd are images, text boxes, Wikipedia content, and Youtube videos, and the result delivered by this system is an enriched video, that consists of the original video presented synchronized with the extra content provided and selected by the crowd.

The approach taken to achieve the complex annotation needed to enrich the videos is to cascade microtasks that collect simple annotations, instead of collecting complex annotations for each contribution. In this way, people without specialization or training can contribute to the process.

- **Task 1 - Identify the points of interest** in the video that should be associated with the extra content. The first microtask is to send video segments to the worker and ask him to identify in this segment something that he believes deserves to be highlighted or supplemented. The aggregation rules for this microtask are to temporarily group the annotations with a tolerance of 0.5 sec, to count and to merge similar annotations in each group, and to determine for each time group which is the predominant point of interest in the annotations.
- **Task 2: Provide extra content suggestions** for each point of interest. In the second task, the worker receives a point of interest and should suggest extra content related to it. This content can be a text, an image, a YouTube video or a Wikipedia page. The aggregation of the second task consists in grouping the contributions by a point of interest and joining similar contributions to avoid duplicity.
- **Task 3: Ranking the suggested content** provided by each point of interest. In the third microtask, the worker receives a point of interest and the content suggestions for it. The contributor should choose the most appropriate content for the point presented. The aggregation rule for this task is to select the most popular content for each point of interest.
- **Task 4: Determine the positions** to display the extra content associated with each point of interest. In this task, the worker receives an item that represents a point of interest and chooses the position in the video most suitable to display it. The aggregation method for this task calculates the average coordinate for each item to be displayed in the video.

5.1 Cascading Microtasks

The adopted approach consists of dividing the complex annotation into simple annotations that can be collected by a set of simple annotation tools. Each of these simple annotations is collected by a microtask.

As is illustrated in Figure 11, the input for each task is generated by the Aggregator after the previous task, except for the task 1. For this task is provided a bootstrap Input that is a list of video segments provided by the owner, that is who initiate the process. Each entry of the bootstrap input can represent a semantic block of the video.

Other applications that use CrowdNote may use different strategies to segment videos such as fixed time-length, SRT files, or even add a microtask to segment videos.

5.2 Task 1

Identify Points of Interest: The first annotation microtask is supported by the tool represented in Figure 12, collecting identification for points of interest. In this task, the contributor receives

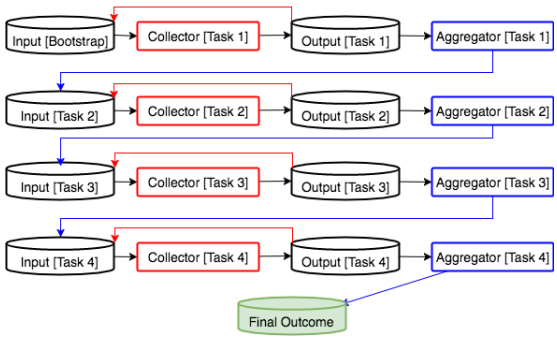

Figure 11: Cascading Microtasks

a segment of video that should be watched, and if was found any point of interest, it should be marked and briefly described. These points of interest can be gestures, words, expressions, facts, concept, characters, events or anything that can be related to extra content.

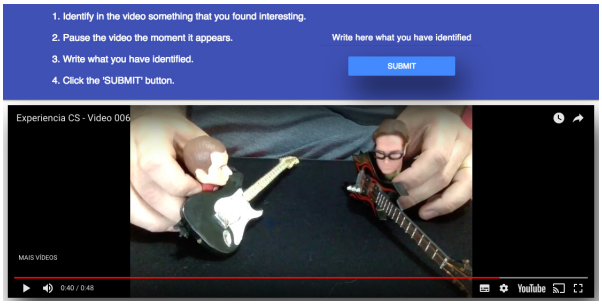

Figure 12: Annotation Tool for Task 1

5.3 Task 2

Provide extra content suggestions: The second task took as input the aggregated result from the task 1 that is a list of points of interest identified by the workers. This microtask is supported by the annotation tool represented in Figure 13. This tool presents to the worker a point of interest and the video segment positioned at the moment it occurs. This way, is possible to use the video for reference and contextualization.

Through this tool, the worker can contribute by writing a text related to the point of interest, sending an image or sending a link to a YouTube video or a Wikipedia page.

When the collection of contributions for this task is done, the Aggregator groups the content of the sender by a point of interest, and then joins the similar suggestions. In this way, a list of points of interest with a set of content suggestions for each is added to the next task, without repeated suggestions.

5.4 Task 3

Ranking Suggestions: The third task receives as input the list of points of interest, with the content suggestions for each of them. For each job, the annotation tool illustrated in Figure 6 shows the worker a point of interest and the video positioned at the time that

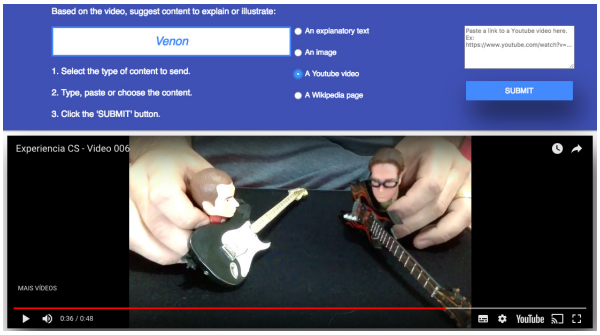

Figure 13: Annotation Tool for Task 2

point occurs. The annotation tool displays the content suggestions for that point of interest below the video, so is possible to browse through the content to choose the most appropriate one.

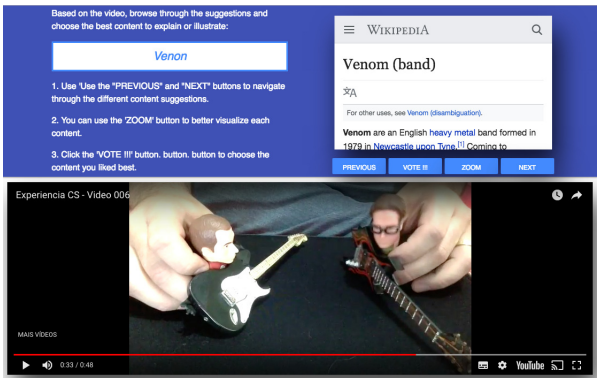

Figure 14: Annotation Tool for Task 3

The worker can enlarge each content to see it better, how can be seen in Figure 15. In addition to playing the videos as a suggestion of content.

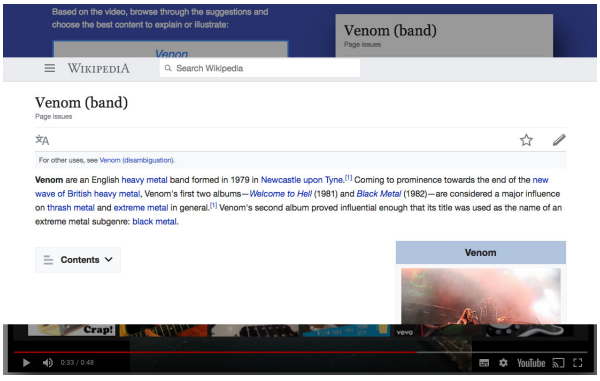

Figure 15: Annotation Tool for Task 3 - Zoom

The aggregation process for this task counts the votes for each content suggestion and chooses the most popular content for each point of interest.

5.5 Task 4

Determine the positions: The last task receives as input the list of points of interest and the content chosen to associate with it. For each job, the tool shown in Figure 16 shows the worker the video that is positioned at the time the point of interest occurs and the reference item for the content selected in the video.

The contribution to this task is to suggest the best position to present the extra content, using the annotation tool to determine this position. The tool allows the worker to change the position of the items in the video by clicking the desired point. Among the 4 microtasks, this is the fastest and easiest to perform.

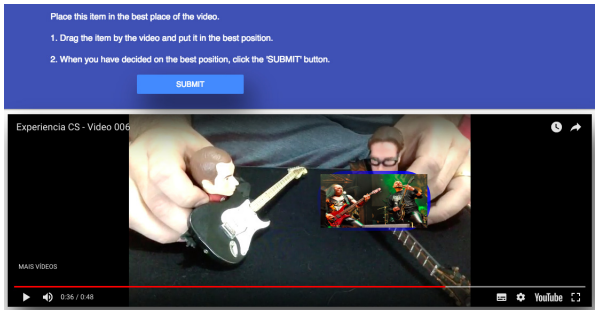

Figure 16: Annotation Tool for Task 4

Following the studies about the wisdom of the crowd, the strategy to determine the correct position is to calculate the average coordinate of the contribution for each content [19]. In this way, the aggregation process calculates the average coordinate of the items, based on the contributions of the crowd. The result of this process is the position where each item related to a point of interest will appear in the video.

5.6 Player

The presentation system, shown in Figure 17, receives the video, extra content, and necessary metadata from the Player Provider. This system is capable of reproducing the original video synchronized with the extra content, that is displayed every time a point of interest happens in the video. Is important to remind that all extra content displayed with the video was provided, selected and positioned by the crowd.

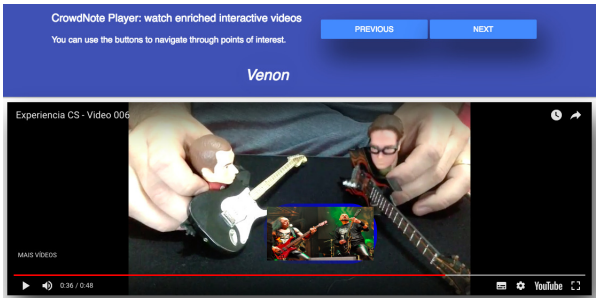

Figure 17: Displaying an extra content item over the video

When the user clicks on some extra content displayed in the video, the presentation is paused and a larger preview of the selected content is displayed in a zoom box. This system features navigation by extra-content instead of the traditional timeline navigation, making available a button-bar with buttons to navigate among the extra contents.

6 FINAL REMARKS

This paper introduced a crowdsourcing approach to annotate videos without requiring experts, trained workers or time-consuming tasks. Moreover, was conducted an experiment to validate it by generating interesting annotated videos that could be used to create interactive multimedia presentations. To support this experiment was developed a toolkit that includes the presentation system, and a set of video annotation tools and aggregation methods.

During the annotation stage, it was noticed that the faster microtasks received more contributions, because the workers contributed more times, annotating more items. One conclusion about this is that volunteers use to dedicate a set time to perform tasks, so they were willing to execute any number of microtasks during that interval.

Another observation about the approach is that the cascade of tasks results in the generation of partial results that can be used for other purposes. For example, content suggestions that have been collected to annotate the video can be used to populate an online dictionary or encyclopedia.

Moreover, the individual aggregation of the result of each microtask allows an adequate processing for each annotation, as well as specific validations for them.

Perhaps one of the most interesting results was to see if this approach is capable of generating systems that can be reused and expanded. This can be observed when the first presentation system was generated and later a new task was added in the process, allowing the construction of an improved presentation system.

In addition to the approach presented, which was able to guide crowdsourcing annotation processes with a certain degree of complexity, a system was also generated that demonstrates how this approach can be applied. This system is available for use and can be used both to replicate this experience and to perform other works.

6.1 Next Steps

An immediate improvement in the system includes changes in the aggregation methods of tasks 1 and 2. Currently, the similarity comparison uses simple syntactic techniques for content analysis. However, a method is being developed that performs these comparisons through morphosyntactic analysis.

The owner module will also be developed, which will allow this system to be used even outside the academic environment. Currently, the system counts only as microtask execution module, which was necessary to perform the experiment.

This work also served as a starting point for a series of projects that will be developed shortly. In particular, the approach presented will be refined to become a complete method.

ACKNOWLEDGMENTS

The authors would like to thank FAPES, CAPES and CNPq for financial support of this research.

REFERENCES

[1] 2014. *VidWiki: Enabling the Crowd to Improve the Legibility of Online Educational Videos.* ACM Conference on Computer Supported Cooperative Work. https://www.microsoft.com/en-us/research/publication/vidwiki-enabling-the-crowd-to-improve-the-legibility-of-online-educational-videos/

[2] 2017. (2017). https://play.google.com/store/apps/details?id=com.google.android.apps.village.boond

[3] Geovanny Abaunza and M. José Rodríguez-Conde. 2016. Bibliographic Review on Web Applications Used to Learn a Foreign Language. In *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM '16)*. ACM, New York, NY, USA, 229–234. https://doi.org/10.1145/3012430.3012522

[4] Luis von Ahn. 2011. Massive-scale online collaboration. (Apr 2011). http://www.ted.com/talks/luis_von_ahn_massive_scale_online_collaboration

[5] Marcello Amorim, Ricardo Mendes, Celso Alberto Saibel Santos, and Orivaldo Tavares. 2017. CrowdNote: Crowdsourcing Environment for Complex Video Annotations. In *XVI WFA ()*. http://XXXXX/175831.pdf

[6] Marcello Amorim, Celso Alberto Saibel Santos, Ricardo Mendes, and Orivaldo Tavares. 2017. Video Annotation by Cascading Microtasks: a Crowdsourcing Approach. In *WebMedia 2017 - Full and Short papers ()*. Gramado, RS. https://doi.org/10.1145/3126858.3126897

[7] Marco Bertini, Alberto Del Bimbo, Andrea Ferracani, Francesco Gelli, Daniele Maddaluno, and Daniele Pezzatini. 2013. Socially-aware video recommendation using users' profiles and crowdsourced annotations. In *Proceedings of the 2nd international workshop on Socially-aware multimedia*. ACM, 13–18.

[8] Joan-Isaac Biel and Daniel Gatica-Perez. 2013. The youtube lens: Crowdsourced personality impressions and audiovisual analysis of vlogs. *Multimedia, IEEE Transactions on* 15, 1 (2013), 41–55.

[9] Chen Chen, Xiaojun Meng, Shengdong Zhao, and Morten Fjeld. 2017. ReTool: Interactive Microtask and Workflow Design Through Demonstration. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 3551–3556. https://doi.org/10.1145/3025453.3025969

[10] Si Chen, Muyuan Li, Kui Ren, and Chunming Qiao. 2015. Crowd map: Accurate reconstruction of indoor floor plans from crowdsourced sensor-rich videos. In *Distributed Computing Systems (ICDCS), 2015 IEEE 35th International Conference on*. IEEE, 1–10.

[11] Bruna C.R. Cunha, Rodolfo Dias Correia, and Maria da Graça Campos Pimentel. 2015. Mobile Video Annotations: A Case Study on Supporting Rehabilitation Exercises. In *Proceedings of the 21st Brazilian Symposium on Multimedia and the Web (WebMedia '15)*. ACM, New York, NY, USA, 245–252. https://doi.org/10.1145/2820426.2820449

[12] Marcello N de Amorim, Ricardo MC Segundo, and Celso AS Santos. 2016. LiveSync: a Tool for Real Time Video Streaming Synchronization from Independent Sources. In *WebMedia 2016 WFA*. Teresina - PI, Brazil.

[13] Travis Desell, Kyle Goehner, Alicia Andes, Rebecca Eckroad, and Susan Ellis-Felege. 2015. On the effectiveness of crowd sourcing avian nesting video analysis at Wildlife@ Home. *Procedia Computer Science* 51 (2015), 384–393.

[14] Rucha Deshpande, Tayfun Tuna, Jaspal Subhlok, and Lecia Barker. 2014. A crowdsourcing caption editor for educational videos. In *Frontiers in Education Conference (FIE), 2014 IEEE*. IEEE, 1–8.

[15] Djellel Eddine Difallah, Michele Catasta, Gianluca Demartini, Panagiotis G. Ipeirotis, and Philippe Cudré-Mauroux. 2015. The Dynamics of Micro-Task Crowdsourcing: The Case of Amazon MTurk. In *Proceedings of the 24th International Conference on World Wide Web (WWW '15)*. ACM, New York, NY, USA, 238–247. https://doi.org/10.1145/2736277.2741685

[16] Guilherme Fião, Teresa Romão, Nuno Correia, Pedro Centieiro, and A. Eduardo Dias. 2016. Automatic Generation of Sport Video Highlights Based on Fan's Emotions and Content. In *Proceedings of the 13th International Conference on Advances in Computer Entertainment Technology (ACE2016)*. ACM, New York, NY, USA, Article 29, 6 pages. https://doi.org/10.1145/3001773.3001802

[17] Bauke Freiburg, Jaap Kamps, and Cees GM Snoek. 2011. Crowdsourcing visual detectors for video search. In *Proceedings of the 19th ACM international conference on Multimedia*. ACM, 913–916.

[18] Neeraj J Gadgil, Khalid Tahboub, David Kirsh, and Edward J Delp. 2014. A web-based video annotation system for crowdsourcing surveillance videos. In *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 90270A–90270A.

[19] FRANCIS GALTON. 1907. Vox Populi (The Wisdom of Crowds). *Nature* 75, 1949 (1907), 450–451. https://doi.org/10.1038/075509f0

[20] Dan B. Goldman, Chris Gonterman, Brian Curless, David Salesin, and Steven M. Seitz. 2008. Video Object Annotation, Navigation, and Composition. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology (UIST '08)*. ACM, New York, NY, USA, 3–12. https://doi.org/10.1145/1449715.1449719

[21] Luke Gottlieb, Jaeyoung Choi, Pascal Kelm, Thomas Sikora, and Gerald Friedland. 2012. Pushing the limits of mechanical turk: qualifying the crowd for video geolocation. In *Proceedings of the ACM multimedia 2012 workshop on Crowdsourcing for multimedia*. ACM, 23–28.

[22] Daniel Haas, Jason Ansel, Lydia Gu, and Adam Marcus. 2015. Argonaut: Macrotask Crowdsourcing for Complex Data Processing. *Proc. VLDB Endow.* 8, 12 (Aug. 2015), 1642–1653. https://doi.org/10.14778/2824032.2824062

[23] Chul-Hee Han and Jong-Seok Lee. 2014. Quality assessment of on-line videos using metadata. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 1385–1388.

[24] Jisup Hong and Collin F. Baker. 2011. How Good is the Crowd at "Real" WSD?. In *Proceedings of the 5th Linguistic Annotation Workshop (LAW V '11)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 30–37. http://dl.acm.org/citation.cfm?id=2018966.2018970

[25] M. Hosseini, K. Phalp, J. Taylor, and R. Ali. 2014. The four pillars of crowdsourcing: A reference model. In *2014 IEEE Eighth International Conference on Research Challenges in Information Science (RCIS)*. 1–12. https://doi.org/10.1109/RCIS.2014.6861072

[26] Jeff Howe. 2006. The Rise of Crowdsourcing. *Wired Magazine* 14, 6 (06 2006). http://www.wired.com/wired/archive/14.06/crowds.html

[27] Samuel Huron, Petra Isenberg, and Jean Daniel Fekete. 2013. PolemicTweet: Video Annotation and Analysis through Tagged Tweets. In *Human-Computer Interaction–INTERACT 2013*. Springer, 135–152.

[28] Hernisa Kacorri, Kaoru Shinkawa, and Shin Saito. 2014. Introducing game elements in crowdsourced video captioning by non-experts. In *Proceedings of the 11th Web for All Conference*. ACM, 29.

[29] Gunhee Kim, Leonid Sigal, and Eric P. Xing. 2014. Joint Summarization of Large-Scale Collections of Web Images and Videos for Storyline Reconstruction. In *Proceedings of the 2014 IEEE CCVPR (CVPR '14)*. IEEE Computer Society, Washington, DC, USA, 4225–4232. https://doi.org/10.1109/CVPR.2014.538

[30] Edith Law and Luis von Ahn. 2011. Human Computation. *Synthesis Lectures on Artificial Intelligence and Machine Learning* 5, 3 (2011), 1–121. https://doi.org/10.2200/S00371ED1V01Y201107AIM013 arXiv:http://dx.doi.org/10.2200/S00371ED1V01Y201107AIM013

[31] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature* 521, 7553 (2015), 436–444.

[32] Rada Mihalcea and Andras Csomai. 2007. Wikify!: Linking Documents to Encyclopedic Knowledge. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management (CIKM '07)*. ACM, New York, NY, USA, 233–242. https://doi.org/10.1145/1321440.1321475

[33] Melanie Misanchuk and Tiffany Anderson. 2001. Building Community in an Online Learning Environment: Communication, Cooperation and Collaboration. (2001).

[34] Luyi Mo, Reynold Cheng, Ben Kao, Xuan S. Yang, Chenghui Ren, Siyu Lei, David W. Cheung, and Eric Lo. 2013. Optimizing Plurality for Human Intelligence Tasks. In *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management (CIKM '13)*. ACM, New York, NY, USA, 1929–1938. https://doi.org/10.1145/2505515.2505755

[35] Venkatesh N Murthy, Subhransu Maji, and R Manmatha. 2015. Automatic image annotation using deep learning representations. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*. ACM, 603–606.

[36] Sunghyun Park, Philippa Shoemark, and Louis-Philippe Morency. 2014. Toward crowdsourcing micro-level behavior annotations: the challenges of interface, training, and generalization. In *Proceedings of the 19th international conference on Intelligent User Interfaces*. ACM, 37–46.

[37] José Pedro Pinto and Paula Viana. 2013. TAG4VD: a game for collaborative video annotation. In *Proceedings of the 2013 ACM international workshop on Immersive media experiences*. ACM, 25–28.

[38] Laurel D Riek, Maria F OâĂŹconnor, and Peter Robinson. 2011. Guess what? a game for affective annotation of video using crowd sourcing. In *Affective computing and intelligent interaction*. Springer, 277–285.

[39] Stephen Robertson, Milan Vojnovic, and Ingmar Weber. 2009. Rethinking the ESP Game. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems (CHI EA '09)*. ACM, New York, NY, USA, 3937–3942. https://doi.org/10.1145/1520340.1520597

[40] Paul Rohwer. 2010. A Note on Human Computation Limits. In *Proceedings of the ACM SIGKDD Workshop on Human Computation (HCOMP '10)*. ACM, New York, NY, USA, 38–40. https://doi.org/10.1145/1837885.1837897

[41] Dairazalia Sanchez-Cortes, Shiro Kumano, Kazuhiro Otsuka, and Daniel Gatica-Perez. 2015. In the Mood for Vlog: Multimodal Inference in Conversational Social Video. *ACM Transactions on Interactive Intelligent Systems (TiiS)* 5, 2 (2015), 9.

[42] CAS Santos, Alexandre SANTOS, and TA Tavares. 2007. Uma estratégia para a construção de ambientes para a descrição semântica de vídeos. (2007).

[43] Elizeu Santos-Neto, Tatiana Pontes, Jussara M Almeida, and Matei Ripeanu. 2014. Towards Boosting Video Popularity via Tag Selection.. In *SoMuS@ ICMR*. Citeseer.

[44] Robert J. Simmons. 2010. Profile Luis Von Ahn: ReCaptcha, Games with a Purpose. *XRDS* 17, 2 (Dec. 2010), 49–49. https://doi.org/10.1145/1869086.1869102

[45] Fabio Sulser, Ivan Giangreco, and Heiko Schuldt. 2014. Crowd-based semantic event detection and video annotation for sports videos. In *Proceedings of the 2014 International ACM Workshop on Crowdsourcing for Multimedia*. ACM, 63–68.

[46] James Surowiecki. 2005. *The wisdom of crowds*. Anchor.

[47] Luis Von Ahn. 2005. *Human Computation*. Ph.D. Dissertation. Pittsburgh, PA, USA. Advisor(s) Blum, Manuel. AAI3205378.

[48] Luis von Ahn. 2011. Three Human Computation Projects. In *Proceedings of the 42Nd ACM Technical Symposium on Computer Science Education (SIGCSE '11)*. ACM, New York, NY, USA, 691–692. https://doi.org/10.1145/1953163.1953354

[49] Carl Vondrick, Donald Patterson, and Deva Ramanan. 2013. Efficiently Scaling Up Crowdsourced Video Annotation. *Int. J. Comput. Vision* 101, 1 (Jan. 2013), 184–204. https://doi.org/10.1007/s11263-012-0564-1

[50] Meng Wang and Xian-Sheng Hua. 2011. Active Learning in Multimedia Annotation and Retrieval: A Survey. *ACM Trans. Intell. Syst. Technol.* 2, 2, Article 10 (Feb. 2011), 21 pages. https://doi.org/10.1145/1899412.1899414

[51] Meng Wang, Xian-Sheng Hua, Jinhui Tang, and Richang Hong. 2009. Beyond Distance Measurement: Constructing Neighborhood Similarity for Video Annotation. *Trans. Multi.* 11, 3 (April 2009), 465–476. https://doi.org/10.1109/TMM.2009.2012919

[52] Stefan Wilk, Stephan Kopf, and Wolfgang Effelsberg. 2015. Video Composition by the Crowd: A System to Compose User-generated Videos in Near Real-time. In *Proceedings of the 6th ACM MSC (MMSys '15)*. ACM, New York, NY, USA, 13–24. https://doi.org/10.1145/2713168.2713178

[53] Bin Wu, Erheng Zhong, Ben Tan, Andrew Horner, and Qiang Yang. 2014. Crowdsourced time-sync video tagging using temporal and personalized topic modeling. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 721–730.

[54] Jun Zhang, Xiaoming Fan, Jianyong Wang, and Lizhu Zhou. 2012. Keyword-propagation-based Information Enriching and Noise Removal for Web News Videos. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '12)*. ACM, New York, NY, USA, 561–569. https://doi.org/10.1145/2339530.2339620

[55] Yifan Zhang, Xiaoyu Zhang, Changsheng Xu, and Hanqing Lu. 2007. Personalized Retrieval of Sports Video. In *Proceedings of the International Workshop on Workshop on Multimedia Information Retrieval (MIR '07)*. ACM, New York, NY, USA, 313–322. https://doi.org/10.1145/1290082.1290126