

Turning Wisdom and Effort of Crowds into Complex Media Annotation

ABSTRACT

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. Complex media annotation involves annotating different aspects of media objects as well as relating them. This kind of annotation usually is associated with a demanding process that requires experts and elaborated annotation system. This paper presents a method to achieve complex media annotation without requiring complex 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 process workflow for generating complex annotation. To demonstrate the operation of this method, we developed a video enrichment system and carried out an experiment in which the crowd was responsible for executing a set of simple annotation microtasks through simple tools.

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:

. 2017. Turning Wisdom and Effort of Crowds into Complex Media Annotation. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 12 pages. <https://doi.org/10.1145/nnnnnnnn.nnnnnnn>

1 INTRODUCTION

Media annotation consists of supplementing media objects, such as videos, images, and audios, by adding metadata about their content and context, it is also used for 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 [38].

Annotations can be used to highlight key points and add information to contents presented [8], facilitating the creation of media applications for content-based distribution [54], indexing [55], summarization [14], navigation [18], composition [52] and more, through automatic and manual means [32, 50].

Considering the amount of information, the number of interactions, and the expertise needed to generate an annotation, it is classified in this paper as simple and complex. While simple annotations can be acquired with a simple interaction of workers in a microtask, a complex annotation requires the worker to perform a more tedious, difficult, or time-consuming task in which he needs to perform multiple interactions.

Automatic approaches, such as rule-based and deep learning, usually present satisfactory results in generating media annotation, although they require well-structured media objects and extensive examples database [30]. Thus, spontaneous scenarios involving unplanned and non-standard videos, images, and audios may not provide the requirements to apply these automatic techniques for media annotation [35].

Manual media annotation is suitable for these scenarios because it uses human intelligence to handle the tasks. However, this approach can be high-costly because of the potentially high-density of annotation points in the time-based media, as well as the complex nature of some annotation tasks.

Distributed approaches, whether cooperative or collaborative, are an alternative to bypass the high effort required for manual annotation. In a collaborative approach the contributors work together to solve the main problem. In a cooperative approach each contributor solves a part of the main problem to produce a final result [33].

Crowdsourcing allows the execution of a large-scale cooperative process for media annotations, 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 [24]. Moreover, each task can be sent to many contributors, allowing to compare, check and aggregate the contributions, reducing the chance of producing a biased result [17].

There are frequent problems in using a crowdsourcing approach to media annotation such as balancing the relationship between the complexity of the task and the cost. This cost refers to the profile and qualification required from the workers as to the complexity and difficulty of the task delegated to them.

Complex annotation usually requires more complex tasks, demanding some expertise from contributors and are harder and time-consuming to them. Otherwise, simple annotation tasks can be done by microtasks that can be performed easily and quickly by less skilled workers [12].

Thus, as the goal is to use untrained and unskilled workers, the process must be modeled based on simple annotation microtasks.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

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

© 2017 Association for Computing Machinery.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...\$15.00

<https://doi.org/10.1145/nnnnnnnn.nnnnnnn>

The method presented in this paper aims to achieve a complex media annotation without requiring trained workers or experts, by 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 for generating the outcome. This approach presents some characteristics to get around some problems faced in achieving a complex media annotation, including:

- Manual annotation is used to dispense examples bases and restricted conditions.
- The annotations are provided by ordinary contributors rather than experts or trained workers.
- It is based on simple and quick microtasks instead of hard, time-consuming and tedious tasks.
- Simple annotation tools are used instead of complex systems.

Following this method, each simple annotation microtask is modeled here as a process composed of two steps: 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.

This method is an updated version of [REMOVED FOR BLIND REVIEW][38], with improvements that made it more flexible and useful by overcoming some important limitations detected in the previous version. In addition, improvements were made to the framework designed to support the method. The main improvements consist in:

- The method scope has been expanded to any media object, not just videos.
- Support for parallel and serial tasks.
- Support for non-linear process workflows.
- Improved diagrams for the method.
- Improved diagrams for the framework architecture.
- Support to multiply outputs from each microtask.
- Support for automatic, manual and supervised aggregation methods, instead of just automatic ones.
- Aggregation methods have incorporated more features.
- The annotation tools was refined.
- In the experiment was used the commercial crowdsourcing platform Microworkers, instead of an open crowd.

The objective of this work is to present the updated version of the method and verify if it is possible to design and execute a complex media annotation process from it and the improved framework. For this, a video enrichment system was developed, based on structure. This system was used to perform an experiment in which the crowd was responsible for performing the tasks related to the enrichment process. The result of this experiment was based on the base provided by the original video's author according to his expectations for the enriched version of the video.

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 method. Section 5 presents the framework architecture. Section 6 presents the conducted experiment. Finally, section 7 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 [24] to annotate media objects. 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 to all people but they are extremely difficult even for the most powerful and sophisticated software and machines. This type of task has as its characteristics the need for creativity, emotional sensibility, empathy, sarcasm and level of abstraction which, in the present moment, is inherited in the minds of human beings.

There are still cases where automatic methods can accomplish the task, but the necessary resources do not exist as examples of bases or a well-defined set of rules. Some examples classify this type of task as 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 approaching the problems from this point of view, identifying in it what tasks can be automated and which ones require human treatment. Additionally, Human Computation can improve performance by the division of labor because it helps to define tasks that can be executed in parallel [41].

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 [29].

In short, Human Computation is a paradigm that proposes to identify in a problem which tasks require human intelligence and which tasks can be automated. In general, the problems that might benefit by using Human Computation modeling are problems in which it is possible to identify tasks that are very difficult for machines but can be easily completed by humans. This strategy is illustrated in Figure 1.

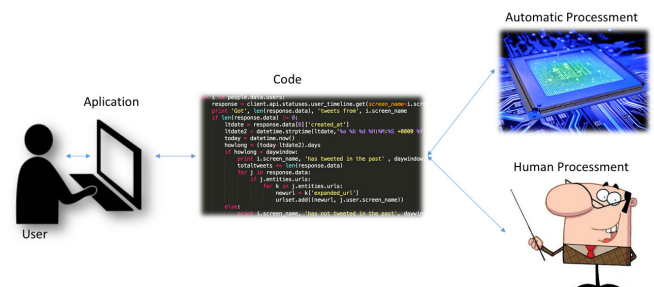


Figure 1: General human computation architecture

A HIT may involve creative activities such as authorship, simple observation as in identifying events in videos, or empathy such as the detection of emotion in facial expression images, but all these tasks can still be performed by people more easily than it would be for the machines. In terms of complexity, a HIT can range from a trivial microtask to a complex macrotask.

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 kind 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 annotation systems and limit the group of workers who are able to execute them [20]. 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. 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 this approach 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 Internet connection speed.

Moreover, considering each microtask as a human computation function, mapping elements from input to output, it is possible to compose an algorithm using these functions to obtain equivalent results generated by macrotasks [5].

2.2 Crowdsourcing

Human computation approaches can improve performance by the division of labor because it helps to execute tasks 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 [41].

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 [24]. Furthermore, each task can be sent to many contributors, allowing to compare, check and to aggregate the contributions, also reducing the chance of producing a biased result [17].

This approach generally delivers good quality results using contributions from a crowd of contributors and can distribute, collect, validate and combine large amounts of tasks results [20, 22, 34].

Since this approach is designed to handle a huge number of cooperators and contributions for tasks that require human intelligence [24], Crowdsourcing is appropriate to allow the Human Computing paradigm to be applied in a massive-scale online cooperation [2].

Crowdsourcing is supported by four pillars: The Crowdsourced Task, The Crowdsourcer, The Crowd, and The Crowdsourcing Platform [23].

- **The Crowdsourced Task** is the HIT designed, according to the Human Computing paradigm, to acquire workers' contributions. Task instances are presented to the workers as jobs that must be performed [12].
- **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 [23].
- **The Crowd** is the workforce 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 [41].
- **The Crowdsourcing Platform** is the centralizer of the whole process, 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 Amazon-Mechanical Turk [12], or really simple systems with screens and forms for data collection such as the mobile application used by the Google Crowdsourcing project [26]. 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 cooperation.

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 gamification.

To exemplify how Crowdsourcing adds support for massive-scale online cooperation to human computation systems, one can analyze the cooperative processes involved in Luis von Ahn's three most well-known projects, reCAPTCHA [44], ESP Game [40], 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 [40]. 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 [1].

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 [17]. 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, an appropriate aggregation method should be applied. These methods may involve the 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 it is required additional human verification or some human interaction in the contributions' processing.
- **Manual:** manual aggregation is used when it's 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 media annotation approaches are used in various applications and are used to gather information of various types, such as temporal synchronization[9, 53], events[28, 45], scene objects[7, 37], emotions[4, 42], actions[10, 39], quality[15, 21], geo-tagging[6, 19], social relevance[3, 25, 43] and captions[11, 27].

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 [4, 7, 10, 19, 25, 36, 49].

VidWiki[7] is a complex system to improve video lesson by annotating them, 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[19] 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[7]

There are also papers on crowdsourcing media annotation that report the use of simple tools and microtasks that can be done quickly by unskilled workers. These works include [5, 9, 16, 28, 37, 39, 45, 53].

The work published by N.Gagil in 2014[16] uses a very simple annotation tool(Figure 3) that allows the workers to perform an easy microtask, which consists of annotating media with surveillance problems if any of them are found.

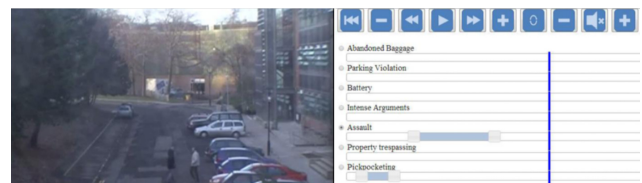


Figure 3: Simple surveillance annotation tool[16]

The work conducted by Dontcheva [13] combined e-learning and crowdsourcing to obtain improved images. The workers can improve their skills in image editing through an interactive step-by-step tutorial and test these skill by improving real-world images submitted by requesters. In the end, the improved images are aggregated to generate an improved image dataset.

4 METHOD

The crowdsourcing media annotation method presented in this paper follows three steps: Preparation, Annotation, and Presentation. In the preparation step, the production process is defined, in the annotation step the contributions are collected and processed and, in the presentation step, the outcome is available for consumption.

4.1 Preparation

All activities involved in this step are performed by the owner, who started the media annotation process. The complex media annotation desired as a result is analyzed and all information required for its construction is identified.

Then it is necessary to determine the simple annotation tasks required to obtain this information and the order of dependency between the tasks. For each of these tasks, it is necessary to define:

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

Once the simple media annotation microtasks are defined, a workflow can be constructed to determine how they fit into the complex annotation production process. The built workflow should follow the general model shown in Figure 4. This workflow will guide the annotation step.

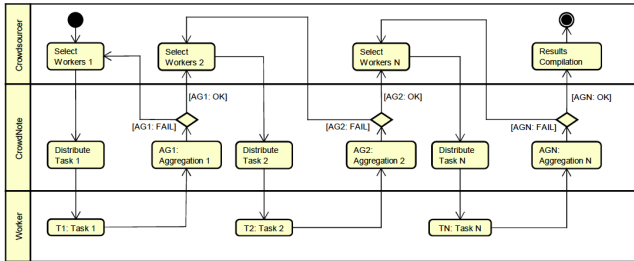


Figure 4: Process workflow

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 image or segment of audio or 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.

4.2 Annotation

The workflow produced in the preparation step defines how the output from a task will be used as input to the next one. In this way, the output of the last task is equal to the desired outcome of the process. In addition, each task can generate multiple outputs that are different representations of the collected and aggregated data. For example, events identified in a video by a task can either serve as input to the next task or generate indices or summaries for the video.

This cascading microtasks workflow is illustrated in Figure 4. It is important to notice that each iteration in this workflow is composed of two activities, the task in self and the aggregation activity, that generates the output from the obtained contributions.

The method determines that annotations are collected from the workers on each of the tasks, however, these annotations can be obtained from crowds or crowdsourcing indoor environments, or from open crowds brought together by an open call.

4.3 Presentation

In the presentation step the result generated is made available, it can be displayed by a player or exported in different formats. In fact, the output can be exported in several formats, because the complex annotation generated is stored as metadata and offers flexibility in this regard.

5 FRAMEWORK

The architecture of the framework designed to support the presented method is represented in Figure 5.

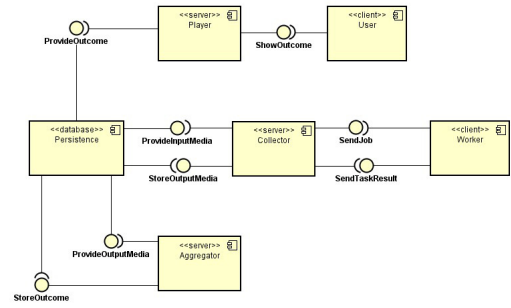


Figure 5: Framework Architecture

The interaction between the system modules through the interfaces are described as follow:

- **Provide Media Input:** To generate each job to be sent to a worker, the Collector receives an entry from the Persistence component.
- **Send Job:** The Collector sends a job to a Worker who sees the task through the Client and executes it.
- **Send Job Result:** The Client sends to the Collector the annotation made by the Worker for the job received.
- **Store Media Input:** The Collector sends the collected contribution to the Persistence that stores it in the Database.
- **Provide Output Media:** The Persistence send to the Aggregator all the collected contributions for a task.
- **Store Outcome:** 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.
- **Provide Outcome:** The Aggregator sends the outcome to Persistence to store it.
- **Show Outcome:** The Player displays to a User the outcome generated by the cascading microtasks process.

The three architecture modules Database, Server, and Client, as well as their components, are described below.

5.1 Database

The database module contains the persistence module that is the point of communication between the modules of the structure. In this way, the modules can be decoupled, increasing the flexibility of the system and the possibilities of integration with external environments in a service-oriented architecture.

The persistence component sends to the server module the information required to render in the client both, the job requests to workers and the content needed to present the result to the users. Once the persistence component communicates with its own database, there is no need for the crowdsourcing platform, if used, to have access to the dataset.

Likewise, contributions provided by workers are sent directly from the collector to the persistence component, so that they are stored directly in the database without having to go through the external crowdsourcing environment in no time

The aggregation component also communicates directly with persistence. The aggregator requests each run the collected contributions to a task and, after the aggregation process, sends the result to be stored from the database to be used as input to the next task, again maintaining data privacy because it does not need to be stored in any external environment.

5.2 Server

The server module is responsible to distribute the jobs, manage contributions, and control the active task in order to execute the process workflow. This module is composed of 3 components: Collector, Aggregator, and Player.

- **Collector:** The collector feeds an annotation tool with the information about the item to be annotated, so it renders the job's interface used by the worker to perform the task. Completely, this component is also responsible for gathering the information provided by the worker in the execution of the task and sends them to the persistence so that they are stored.
- **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 by the persistence component. In this update version, the aggregation process can be fully automatic, supervised or manual. Manual aggregation is very useful when is desired to evaluate each contribution as in authoring tasks. Supervise aggregation can be a good option when is possible to apply automatic methods but is required human verification of the result. Automatic aggregation is the default choice, this class of methods includes grouping, comparing, counting, calculating and other operations.
- **Player:** The player component works in a way very similar to the collector, although it renders the content to users and its communication with the persistence is unidirectional. It feeds the client player tool with the metadata, the extra content, and the original video. Thus, the player tool on the client can play the enriched video synchronously.

5.3 Client

The client module manages the communication interfaces with workers and users. The approach used is to use templates that are fed with the descriptive data of the visualization to be generated.

In this way, it is possible to keep the entire control part on the server, accessible through the collector and player components, and therefore templates can be stored anywhere if it is accessible on the internet. This allows contributions to be collected from crowdsourcing platforms and web pages published on different hosts. In fact, this strategy allows to collect entries from different sources at the same time.

Both the container collection view and the final content view display work in a similar fashion. According to the desired visualization, a visualization model is selected and the server module is queried to obtain the necessary data to render the desired interface. This process is illustrated in Figure 6.

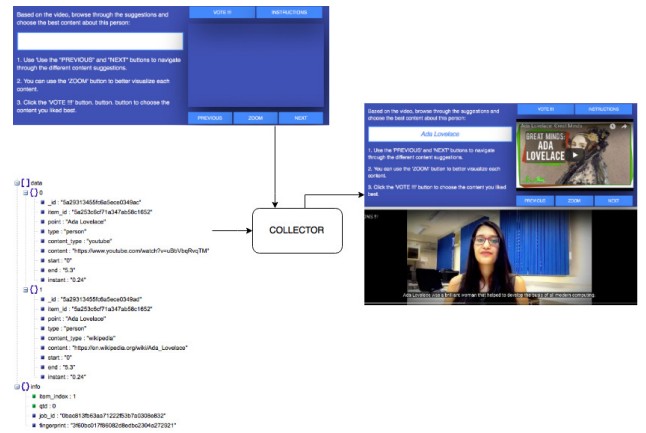


Figure 6: Job rendering

6 CASE STUDY

The experiment used the presented method to enrich videos with additional semantic information about characters, theories, and technologies mentioned. The rich video generated is a self-contained multimedia document, so the user doesn't need to use other sites or databases to access supplementary content for the points of interest.

Each supplemental content can be accessed through trigger items that are images and labels that appear in scenes where the point of interest is mentioned. When the trigger items are clicked, the video pauses the extra content is displayed so they can be viewed, and when closed, video playback resumes normally.

The crowdsourcing process used in the experiment followed a workflow of 4 annotation microtasks. Each of these simple tasks was modeled that could be performed by a worker through a simple tool annotation, only that the worker could understand the language spoken in the video. The video was produced in the English language and features an actress narrating a text about the history of computing.

6.1 Our Crowd

To reach a heterogeneous crowd, the Microworkers [31] platform was used, this platform was used only to recruit and pay workers, and the whole process was supported by the developed system.

Microworkers proposes different models for a crowdsourcing process. Initially, you can choose between starting a basic campaign or using contracted groups.

In a basic campaign, all registered workers in the system can see the task on the job wall and accept to run it. A campaign that uses hired groups allows you to select the crowd by choosing groups of workers with a certain profile. In addition, it is possible to create lists with good workers who have made good contributions to recruit them for other tasks. Some of hired groups available in Microworks are listed in Figure 7.

ALL AU Workers(3650)	Data Services(1153)
ALL Canada Workers(5388)	Designers(10)
ALL European Workers(140698)	English Speaking Countries(207055)
ALL German Workers(1523)	Finland (FI)(945)
ALL UK Workers(7766)	French Speaking Countries(26171)
ALL USA and Canada Workers(62310)	German Speaking Countries(12370)
ALL USA Workers(56922)	Global Users(1055921)
Best Rated Countries(4688)	International Workers(8327)
Bloggers(673)	iOS/Android(1730)
BRICS Countries(54948)	Programmers(91)
Canada Ontario(2524)	SMS-Verified(160325)
Canada Quebec(807)	Spanish Speaking Countries(15448)
Cantonese(116984)	Top AU Workers(54)
Cantonese Writers (Verified)(52)	Top Bangladesh Workers(864)
Caption Writers(37)	Top Canada Workers(118)
Chinese Speaking Countries(7375)	Top Earners(1000)

Figure 7: Microworkers Hired Groups

The purpose of this paper is to use unskilled workers, although it was necessary for them to know enough of the English language to understand the video used in the experiment. In this way, the tasks were launched as campaigns that used contracted groups, to increase the chance of the workers who contributed in a task also to participate in others, was chosen a group of moderate size and with workers relatively assiduous so that the contributions were made more quickly. The group chosen was Data Services, with 1153 potential workers to accept the jobs.

Some groups are made up of workers who only accept tasks that offer slightly larger payments, but considering the group chosen, it was feasible to offer a payment of 2 cents per task. Also, each task was active for 24 hours to reach all time zones in the same way.

6.2 Preparation

The preparation step began with the identification of the microtasks needed to achieve the expected result. Since the goal was to generate enriched videos through supplementary content at points of interest, it was determined that they were needed. Thus, it was determined that 4 microtasks would be performed:

- Identify the points of interest in the video;
- Gather content suggestions for each point of interest;
- Determine the most accepted supplementary content for each point of interest;
- Position the trigger items at the best position over the video.

In relation to the distribution of the items to be annotated in each of the tasks, a circular list policy was adopted. In this way, there was a greater chance of each task, each item being annotated received the same number of contributions.

It was also necessary to determine how the video should be segmented to generate the input of the first task. The strategy chosen was to target the video based on the automatic captions generated by youtube. In this way, it was possible to generate short segments, but they tend to contain complete sentences. This method divided the video into 13 segments.

For budget and time issues, it was determined that the minimum amount of contributions each item could receive was 5, so Task 0 should receive at least 75 contributions. The number 5 was chosen because it is odd, which avoids drawings, and because it is an amount that already allows seeing a tendency of convergence of opinion in a heterogeneous multitude.

6.3 Annotation

In this section will be described the 4 annotation tasks, as well the annotation tools, aggregation methods and results for each of them. Because there is a dependency order between these tasks, it's not possible run them in parallel. In this way, an execution workflow in which the output of one task is used for the next one, as can be seen in Figure 8, in which the tasks are identified as T0 to T3 and their respective aggregation activities as AG0 to AG 3. The payment activities are identified as IA0 to IA3.

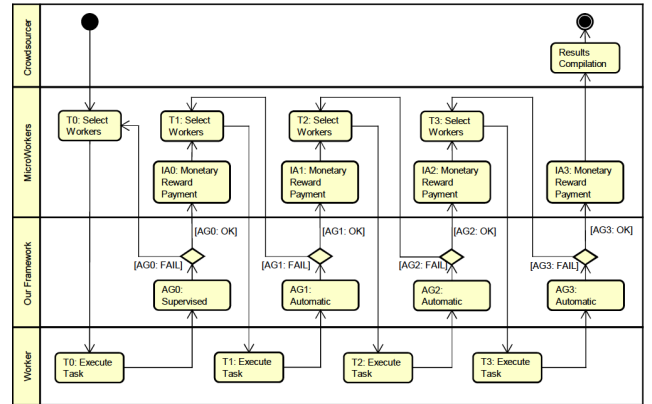


Figure 8: Video Enrichment Workflow

6.3.1 Task 0.

- **Title:** Identify Points of Interest.
- **Description:** A video segment is displayed to the worker and he must identify the moment at which a point of interest appears or is mentioned. This point of interest can be a person, a technology or a theory.
- **Objective:** Identify points of interest in a video segment.
- **Input:** A dataset with 13 video segments of 4.5 to 7 sec.
- **Output:** A set of points of interest and the instant when they happen. In addition to serving as input to Task 1, this result also generated a summary based on points of interest.
- **Instructions:**
 - (1) Identify in the video something that you found interesting.
 - (2) Pause the video the moment it appears.
 - (3) Select its type: Person, Technology or Theory.
 - (4) Write what you have identified.

- **Annotation Tool:** The tool is represented in Figure 9.

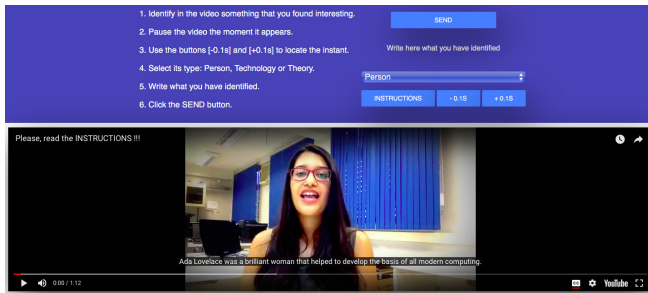


Figure 9: Annotation Tool for Task 0

- **Aggregation:** For this task, a supervised aggregation method was chosen in which the human interaction occurs at the end of the process. The contributions are filtered discarding the useless contributions (empty, nonsense, bad works), and the valid contributions are candidates to Points of Interest. The candidates are grouped in 2-seconds interval ranges. In each group, a similarity analysis is done and the most frequent occurrence is selected. Finally, the human supervisor edits the selected label so that the text is visually pleasing. The aggregation method extracted 11 Points of Interest from the 75 contributions.

6.3.2 Task 1.

- **Title:** Provide Suggestions for Extra Content.
- **Description:** In this task, the worker receives a point of interest and the video synchronized at the moment it occurs. The worker should suggest extra content to complement the video, which may be a short text, an image or a link to a YouTube video or a Wikipedia page.
- **Objective:** Obtain from the crowd extra content to each point of interest.
- **Input:** The set of points of interest collected in Task 0.
- **Output:** A set of extra content associated with each point of interest. Each extra content may be an explanatory text, an image, even a link to a Wikipedia page or a Youtube video. Besides feed the Task 2, this output is also used to generate a content-based index to the video.
- **Instructions:**
 - (1) Select the type of content to send.
 - (2) You can upload an image, write a short text.
 - (3) You also can paste a link to Youtube or Wikipedia.
- **Annotation Tool:** The tool is represented in Figure 10.
- **Aggregation:** For this task was chosen an automatic aggregation method. The filtering step discarded the corrupted files and broken links. The suggested contents were grouped by point of interest, and the duplicated items were merged. The aggregation method extracted 34 different suggestions of extra content from the 55 contributions received. All 11 points of interest received at least 2 content suggestions, some of which received 4 or 5.

6.3.3 Task 2.

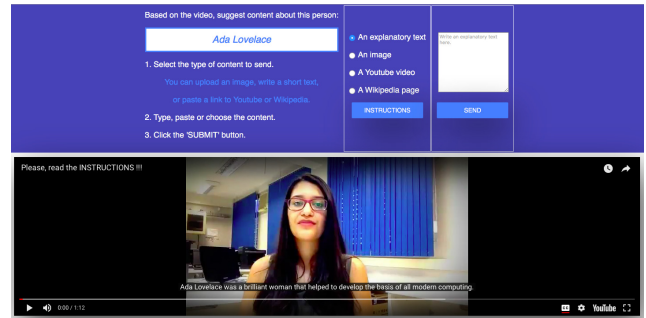


Figure 10: Annotation Tool for Task 1

- **Title:** Ranking Suggestions.
- **Description:** The worker receives a point of interest and the video positioned at the moment it occurs. It also gets the list of suggested extra content to complement this point of interest. The job is to choose the extra content that best complements the content related to the point of interest.
- **Objective:** Determine the best extra content to each point of interest, according to the crowd.
- **Input:** A set of points of interest, with the suggested contents associated with each one.
- **Output:** An updated set of points of interest with the extra content associated with each one. This output also can be used to generate a ranked list of alternative extra contents to supplement the points of interest.
- **Instructions:**
 - (1) Use the buttons bar to navigate through the contents.
 - (2) You can use the zoom button to visualize each content.
 - (3) Vote for the content you liked best.
- **Annotation Tool:** The tool is represented in Figure 11. This tool has a drag-and-drop feature that lets you drag the item through the video until you find the appropriate position.

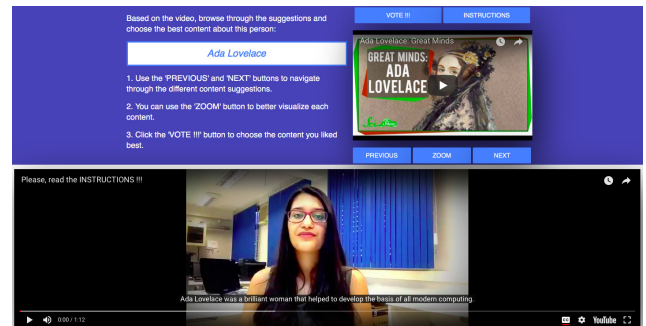


Figure 11: Annotation Tool for Task 2

- **Aggregation:** This simple automatic aggregation method determines the most popular extra content suggested to each point of interest. As the contents were associated with 11 points of interest, were collected 55 contributions. In the filtering step the suggestions without votes was discarded. Finally, was generated an output with all 11 points of interest and the extra content associated with each one.

6.3.4 Task 3.

- **Title:** Determine the Position of the Trigger Items.
- **Description:** This task consists of placing a trigger item in a video scene. This trigger item is represented as a rectangle containing a text or an image and should be positioned so as to minimize occlusions of important scene objects.
- **Objective:** Determine the best position to display each trigger item over the video.
- **Input:** A set of points of interest with the extra content associated with each one of them.
- **Output:** A set of points of interest with the extra content associated with each one, including now the position where each item should be displayed over the video. These positions are represented as coordinates (X,Y). The output of this task is the outcome of the video enrichment process, and can be executed in the Player. In addition, the metadata can be used to generate alternative outputs such as NCL to reproduce them in digital TV environments.
- **Instructions:**
 - (1) Drag the item by the video until finding the best position.
 - (2) When you have decided on the best position, click send.
- **Annotation Tool:** The tool is represented in Figure 12.

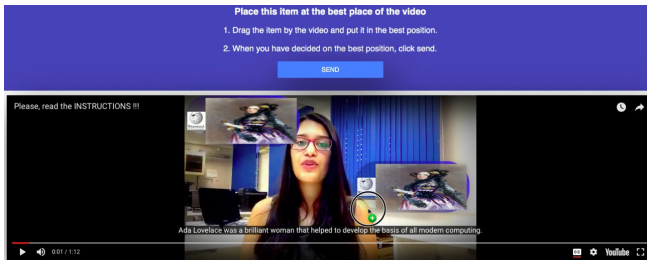


Figure 12: Annotation Tool for Task 3

- **Aggregation:** The automatic aggregation method applied at the final of this task determines the position at each item should be displayed over the video. The contributions for each item were grouped and the very discrepant positions of the others were discarded. Then, for each item, the average geographic position was calculated based on the X and Y coordinates predicted in the contributions. Was collected 55 contributions, of which 38 were after filtering. Of these 38 contributions were extracted the positions of the 11 items related to the points of intersection.

6.4 Presentation

The presentation system, shown in Figure 13, 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.

This tool has a control bar with 3 buttons: Previous, Next and Zoom. These buttons are used to control two useful features, the content zoom and the navigation by point of interest.

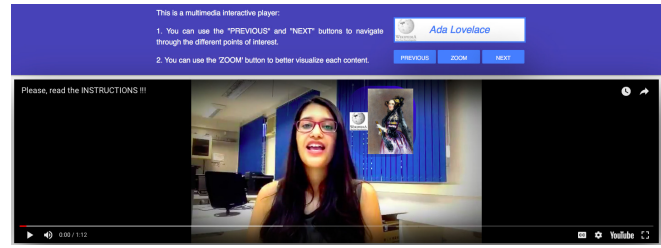


Figure 13: Displaying an extra content item over the video

The navigation by point of interest uses the list of points as a content index, allowing the user to navigate between points of interest by clicking the Previous and Next buttons. The Player automatically syncs the video at the time the current point of interest occurs.

When the user clicks the trigger item or the Zoom button, the extra content associated with the current point of interest is displayed on an upper layer as the video is paused. When you close the extra content, the video resumes its execution from where it was. The zoom view can be seen in Figure 14.

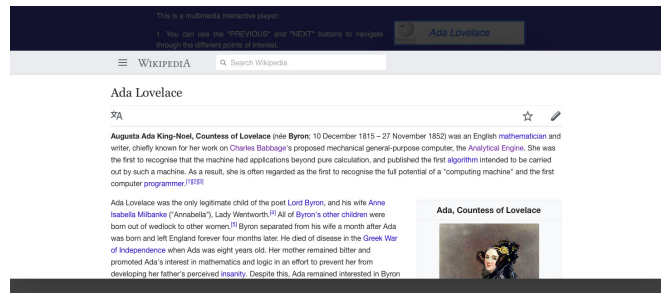


Figure 14: Player Zoom

6.5 Runtime Observations

During some tasks were observed some interesting facts about the behavior of workers. Keeping in mind that the motivation of the workers in the experiment was the payment and that the amount paid for each job was small, the workers tended to do the bare minimum. This aspect proved to be correct that the decision was made to use microtasks that needed only a simple interaction to be completed.

The reflection of this led to some decisions during the experiment, as in Task 1. In this task, the worker was asked to suggest extra content to supplement the points of interest and for this, he could write a short text, paste a link to Wikipedia or Youtube, or upload an image. Most workers provided links and only 2 uploaded images. We thought this was because sending an image was more laborious than pasting a link into the text box, as it was necessary to download the image and upload it.

However, this occurrence was used positively, because few image was obtained in the contributions, we decided to add automatic mechanisms to retrieve them automatically during aggregation. In cases where the selected point of interest was a Wikipedia link mechanism, it retrieved the main page image and, in cases where it was linked to YouTube, the thumbnail was retrieved from the video.

Although this operation is simple, it shows that it is possible to improve the aggregation methods by associating more automatic processing with human contributions. In this way, aggregation methods may be possible integration points with other systems, even with secondary human tasks to create a formal model of supervised aggregation activity.

6.6 Result Evaluation

The result was evaluated according three criteria: points of interest identified, suitability of the associated content, and occlusion of scene objects.

According to the author of the video used in the experiment, there were 21 different points of interest, which after aggregation would result in 14, because at each interval of 2 seconds, only the most relevant point of interest would be selected. The crowd managed to identify 19 points of interest, which after the aggregates generated 11 relevant points. However, the crowd highlighted the term "Formalism" that was not predicted by the author, who after checking it reported that it was valid in the video's context.

The extra content selected for each point of interest was also verified by the author. Of these, only one content did not meet the author's expectations. The content selected for "Logical relations and properties" corresponded to what the author wanted to convey. However, he realized that there was a flaw in the video script and the correct text would be "logical relational and its properties". This way this error can not be attributed to the crowd.

In relation to the occlusion of the objects of the scene, the criterion to calculate items that were positioned in the actress who narrated the text. Of the 11 items, only 2 slightly obscured the outline of the actress, as shown in Figure 15, but no item occluded her image significantly.



Figure 15: Occlusion Issues

In general, the result generated by the crowd was well adapted to convey the content intended by the author, having enriched virtually all the points that he had predicted.

7 CONCLUSION

This paper presented an updated version of [REMOVED FOR BLIND REVIEW], a method for achieving a complex media annotation by running a process workflow composed of simple annotation microtasks in a cascading arrangement. Some relevant improvements were made to both the method and the framework used to support its activities.

The experiment showed that the improvements made in the method were successfully implemented, successfully generating the expected result. The contributions were obtained from a commercial crowdsourcing environment, but with a differentiated approach in which only the resources related to the workers and none of the platform resources were used. This approach ensured that both the data set used and the data collected in the contributions were stored only in our database, not the crowdsourcing platform.

It was also observed that the concept of supervised aggregation introduced in this version obtained positive results. This aggregation approach has proven to be interesting for improving the results of annotation tasks that receive open answers that need to be adjusted manually. A direct conclusion is that this approach can also be used to insert a human verification step at the end of certain aggregation activities.

The diagrams introduced considerably facilitated the creation of the workflow from the production process to the case study. In addition, a specification of the dataflow in the interfaces between the components, facilitating the planning of the distribution of the works and the information that is necessary.

The basic annotation tools distributed with the framework can be easily adapted to work with the commercial crowdsourcing platform as well as to meet the requirements of the tasks required for the experiment.

As can be seen in Figure 16, the workers who contributed to the experiment are scattered all over the world. This ensured a heterogeneous crowd, which contributed to the operation of aggregation methods based on the wisdom theory of the crowds.

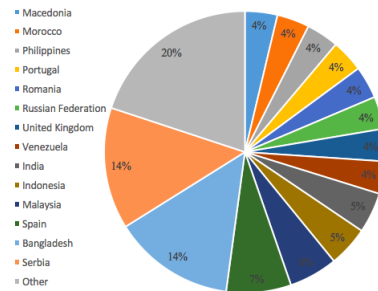


Figure 16: Workers distribution through the world.

The video enrichment process has been able to produce an interactive multimedia presentation from a simple raw video through a crowdsourcing approach. This leads to the conclusion that the updated method presented is appropriate to guide this type of project that aims to generate complex media annotations.

7.1 Issues

Some issues were detected in relation to the collection process. About 20% of the contributions cannot be used because they did not meet the desired specifications or because they had invalid or malicious content. This showed that it is necessary to insert in the annotation tools some resources that induce workers to provide valid contributions. It also alerted us to the importance of giving workers clearer instructions on how to properly perform tasks.

In fact, it drew attention to the need to define criteria for the evaluation of instructions given to workers. Also came the discussion about the need for a research on whether textual instructions are really sufficient to perform any simple annotation microtasking.

7.2 Crowds Comparison Experiment

Another relevant discussion resulting from the observations of the results concerns the extent to which the composition of the crowd can influence the outcome.

To understand the best is not revealed, it is not a presentation method, the experiment conducted will be replicated in different scenarios:

- Increasing the salary of workers;
- Choosing different contracted groups;
- Choosing only the best workers;
- Using a closed group with volunteers;
- Using a group familiar with the subject treated in the video;
- Using a group of natives in the English language.

7.3 Next Steps

This work is constantly improving on three different fronts: improvement of the method, improvement of the framework scenarios and application differences.

Regarding the method, the next steps include the formalization of the aggregation process, defining specific flows for automatic, supervised and manual models.

New annotation tools are being designed for the structure, which can be adapted for more tasks. In addition, a wizard is being designed to guide the creation of complex media annotation projects based on the method presented.

Some experiments are being prepared to be carried out, some of which seem especially promising:

- Multisensorial video annotation for mulsemmedia applications.
- Gesture segmentation in signal language video datasets.
- Semantic approximation of idiomatic expressions in automatic translations.
- Crowdsourcing creation of learning object.

ACKNOWLEDGMENTS

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

REFERENCES

- [1] 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>
- [2] Luis von Ahn. 2011. Massive-scale online collaboration. (Apr 2011). http://www.ted.com/talks/luis_von_ahn_massive_scale_online_collaboration
- [3] 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.
- [4] 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.
- [5] 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>
- [6] 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.
- [7] Andrew Cross, Mydhili Bayyapuredi, Dilip Ravindran, Edward Cutrell, and William Thies. 2014. VidWiki: enabling the crowd to improve the legibility of online educational videos. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*. ACM, 1167–1175.
- [8] 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, 245–252. <https://doi.org/10.1145/2820426.2820449>
- [9] 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*.
- [10] 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.
- [11] 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.
- [12] Djelle 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>
- [13] Mira Dontcheva, Robert R. Morris, Joel R. Brandt, and Elizabeth M. Gerber. 2014. Combining Crowdsourcing and Learning to Improve Engagement and Performance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 3379–3388. <https://doi.org/10.1145/2556288.2557217>
- [14] 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>
- [15] 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.
- [16] 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.
- [17] FRANCIS GALTON. 1907. Vox Populi (The Wisdom of Crowds). *Nature* 75, 1949 (1907), 450–451. <https://doi.org/10.1038/075090f0>
- [18] 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>

- [19] Luke Gottlieb, Jaeyoung Choi, Pascal Kelm, Thomas Sikora, and Gerald Friedland. 2012. Pushing the limits of mechanical turk: qualifying the crowd for video geo-location. In *Proceedings of the ACM multimedia 2012 workshop on Crowdsourcing for multimedia*. ACM, 23–28.
- [20] Daniel Haas, Jason Ansel, Lydia Gu, and Adam Marcus. 2015. Argonaut: Macro-task Crowdsourcing for Complex Data Processing. *Proc. VLDB Endow.* 8, 12 (Aug. 2015), 1642–1653. <https://doi.org/10.14778/2824032.2824062>
- [21] 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.
- [22] 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>
- [23] 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>
- [24] Jeff Howe. 2006. The Rise of Crowdsourcing. *Wired Magazine* 14, 6 (06 2006). <http://www.wired.com/wired/archive/14.06/crowds.html>
- [25] Samuel Huron, Petra Isenber, and Jean Daniel Fekete. 2013. PolemicTweet: Video Annotation and Analysis through Tagged Tweets. In *Human-Computer Interaction-INTERACT 2013*. Springer, 135–152.
- [26] Google Inc. 2017. Google Crowdsourcing project. (2017). <https://play.google.com/store/apps/details?id=com.google.android.apps.village.boond>
- [27] 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.
- [28] 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 CVPR (CVPR '14)*. IEEE Computer Society, Washington, DC, USA, 4225–4232. <https://doi.org/10.1109/CVPR.2014.538>
- [29] 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>
- [30] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature* 521, 7553 (2015), 436–444.
- [31] Microworkers.com. 2017. Microworkers: work and earn, or offer a micro job. (2017). <https://ttv.microworkers.com>
- [32] Rada Mihalcea and Andras Csoma. 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] REMOVED FOR BLIND REVIEW. [n. d.]. REMOVED FOR BLIND REVIEW.
- [39] Laurel D Riek, Maria F O'Ázcon, 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.
- [40] 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>
- [41] 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>
- [42] 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.
- [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>