Measuring Social Distance in Images Using An AI- Human Team

Mohannad Al Ameedi

September 25, 2020

Measuring Social Distance in Images Using An AI- Human Team

Mohannad Al Ameedi
Virginia Tech – Computer Science
email: mohad4@vt.edu

Abstract

Coronavirus was announced as a pandemic by the World Health Organization and affected almost all countries with millions of confirmed cases and hundreds of thousands of deaths. There is no vaccination or treatment to the date of writing this paper, and there are guidelines provided by the WHO and CDC to stay home and keep social distance while being out. Our work is focusing on social distance to help people staying safe and healthy and protect themselves from being infected by the virus. The system provides a search engine that offers information about the virus with image search that label images based on the social distance guideline. People can take precaution steps when they go out for shopping or working on essential role. We use information retrieval to fetch data from news agencies, augment these images from google images. We use AI service to determine if images have people, and human computation [1] via Amazon Mechanical Turk (MTurk) to help with labeling the images according to the social distance guidelines. Users can check these images before they go to grocery stores or work.

1 Introduction/Motivation

Coronavirus spread around the world with hundreds of thousands of confirmed cases and thousands of deaths. There is no effective treatment to the virus or vaccination to prevent the infection yet [2]. Many countries around the world issued stay home orders asking their citizens to only leave their houses when necessary. Leaving home for essential work or shopping might introduce a risk of infection. Our work is trying to help protecting people when they leave their houses by providing visual guidance about social distances. We aim to use information retrieval to crawl news articles and images about coronavirus and then use computer vision and artificial intelligence to classify images that have people. The system uses human computation to help labeling the images [3] based on the distances between any two persons. If people want to go to a grocery store, then they can check images of grocery stores to get guidance of the maximum people that can stay on a single aisle. Similarly, on the workplace, people can look at images to see if two employees can sit next to each other or they need to skip a cube. Although 6 feet distance is easy to be maintained by any individual without the need to see images, but our work will give idea of things that people might expect outside. Our work can be used to provide help with social distance but can also be used as a generic approach to any global event where international organizations issue warnings and guidance and ask people to follow the guidance to reduce the number of casualties. The idea is to use search engine to crawl data about any a global warning, use computer vision to help identify images that apply to the warning, use human computation to help give a precise information about the issue.

2 Related Work

There are similar works that was done to using artificial intelligence and computer vision with human computation to calculate distances in images. There are also work about using human computation with information retrieval. There is a very recent work that was done in New York City about measuring social distancing in sidewalk images on map.

2.1 Popup: Reconstructing 3D Video Using Particle Filtering to Aggregate Crowd Responses [4]

The authors use crowd workers to annotate frames from 2D videos to reconstruct 3D information. They use crowdsourcing to leverages content diversity to collect and aggregate annotation.

The approach is like our work on working on images to get prices measurement, but the goal of the work is different as it is used to reconstruct 3D information.

2.2 Combining Crowdsourcing and Google Street View to Identify Street-level Accessibility Problems [5]

The authors ask crowd workers to label sidewalk accessibility problems using Google Street View imaginary. In this work the authors ask two group of workers, the first group consist of expert sidewalk evaluators, and the second group consist of untrained worker that don not necessary have experience with sidewalk accessibility assessment.
This approach is similar to our work on labeling images to be used by people for their own safety, but we use information retrieval to get the images and use the computer vision to filter the images. Our work goal can be implemented to give guideline to all people in the world.

2.3 Zensors: Adaptive, Rapidly Deployable, Human-Intelligent Sensor Feeds [6]

The authors ask crowd workers to answer sensor reading in nearly instant time without prior training to enable easy creation of sensor in visually observable property.
While the approach is different from our approach, but the concept can be applied by companies to notify their employees if social distance guidelines is not followed or to place light sensors that can tell their customers if a certain areas are not safe from social distance perspective.

2.4 Page Hunt: Improving Search Engines Using Human Computation Games [7]

The authors ask crowd workers to improve search results by helping with ranking the result. The approach is similar to our work in combining search engine with human computation, but we are not concerned about the ranking in our project, as we are interested more in labeling the images that we obtain during the crawling phase.

2.5 Mapping the Sidewalks of New York City for Social Distancing [8]

The authors use map to show the users the sidewalks that are safe to keep social distance. The authors assign colors to sidewalks to according to the width of the sidewalk.

Our approach is different than above approaches as it is combining information storage retrieval, computer vision, and human computation to provide guidance to help people to stay safe. The guidance can be changed according to the severity of the spread of the virus, and artificial intelligence will need time to collect samples and training dataset to give some guidelines but it will be difficult to determine if images are complaint with the guidelines. This approach can be used in any generic settings where you can search for a specific issue, let computer vision determine images that are related to the issue, and get help from human computation to label the images or even text or voice, and present the results to the users.

3 System Description

3.1 Hight level approach

The goal of the system is to help people stay safe and healthy when they are in public place. The system display images to the users with guidance about social distancing by showing them images that adhere to social distance, images that don't adhere to social distance rules, and images that are not applicable. help people stay safe and healthy. Our system uses information retrieval to retrieve images, computer vision to analyze images, human computation to label images.

Information Retrieval [9]**:** We built a crawler that crawl images from different news agency in different categories. We also crawl news articles from these news agencies. All images retrieved are related to coronavirus.

AI: We used computer vision to analyze images to find the number of persons inside the image. We use Microsoft Computer Vision service to accomplish this task. The AI part can help with filtering out images that don't have people and that will help to make the application more scalable and reduce the cost and time to label images.

Human Computation: We used Amazon Mechanical Turk (MTurk) to help with our human computation task. The MTurk worker helped us with labeling the data according to the social distance rules.

Image Hashing: We used image hashing to help us with finding duplicate images so we don't have to send these images to MTurk to save time and money and also to show a consistent result to the users.

User: Users will use our system to see images that related social distancing. Users will login to the system and use text search to search for news articles that related to coronavirus, use image search to see images that are also related to coronavirus. These images are related to grocery stores, parks, doctor offices, workplace, malls, and other places.

These images can be used to as a training dataset for future research about social distancing.

3.2 User experiences and scenarios:

As stated above, users can provide search query to search for images labeled by humans and can be useful in different scenarios. Most countries and cities asked people to stay home and only go out if they need when necessary to go to grocery stores, report to work, or go to the doctors and other essential business, and below are more details about these scenarios:

Grocery stores: users can look at labeled images of grocery stores to get safety guidelines. For example, if different images shows that more than three people on the same aisle is not safe, then users don't have to worry about keeping the distance from other people and can just avoid the aisle and go to a different one until it get less people.

Workplaces: The system can also be used in workplace where employees who must go to work to do essential business can check labeled images related to workplace. They can plan ahead of time to not sit next to their coworker cube as they might never measure the prices distance between two chairs setting in two adjacent cubes. That also apply to meeting room or restroom areas.

Doctor offices: If someone is sick, then the risk of getting infection is high. Seeing labeled images of people in the doctor offices or hospitals can help users staying safe by planning the way they will sit in the waiting area or even when they greet the doctor.

3.3 System Implementation

Figure (1) shows the system architecture and different components that the systems has. The system is built on the top of a previous work by the authors and modified to search for coronavirus news and images only.

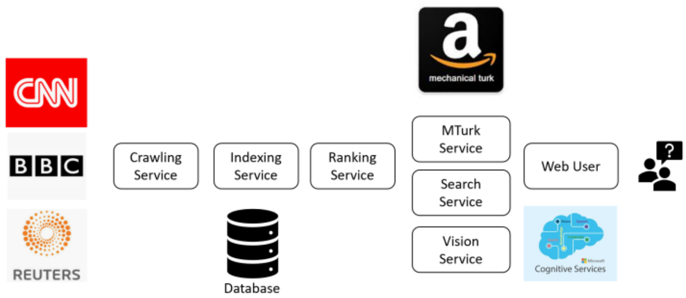


Figure (1) System Architecture of Coronavirus Search Engine.

3.3.1 Search Engine

The search engine crawls news articles and images from popular news agencies RSS feeds. The news agencies are:

- ABC News
- CNN
- Fox News
- BBC News
- Routers
- NYTimes
- Associated Press

The search engine crawls data from different categories which is very important for the social distance labeling as it covers images across different areas. The categories are:

- Health
- Science
- Politics
- Business
- Sport
- Education
- Travel
- World

Crawling Service: The crawling service run on the background to fetch the news articles and their associated images. After fetching the data, the service preprocesses the data to remove stop words and convert all the text to lower case to reduce the index space. The crawling service create an instance of the crawler for each news agency and run in asynchronous mode to speed up the crawling process. The crawling service save the clean data into multiple database tables.

Indexing Service: The indexing service read each news articles and calculate the frequency of each word in each document and then store the results in a term document matrix. The term document matrix table is a mapping table that map each term to the document that contains the term or vocabulary. The indexing service is one of the most time-consuming service in the system and it work on the background as an offline process.

Ranking Service: The ranking service receive the query and partitions it into different keywords, and then contact the indexing service to get a list of documents that match the query in the inverted index. The inverted index is loaded into the main memory to provide results at a constant time. The ranking service uses Term Frequency and Invert Document Frequency (TF/IDF) [10] to rank the documents.

Search Service: The search service receives the search query and process it to remove stop words and convert the query to lower case and contact the query service to get the list of ranked documents and send it to the user interface to be displayed to the user.

3.3.2 Vision Service

This the Artificial Intelligence part of the system and it is used to filter images that have people. We considered both Microsoft Face API and Microsoft Vision AP to label the images. We found that the Face API is not performing well and couldn't recognize people on the following scenarios:

- People faces are not showing in the picture as shown in Figure (2)
- People are wearing masks as shown in Figure (3)

Figure (2) shows an image that has people but was not recognized by Microsoft Face API since their faces are not shown.

Figure (3) shows an image that has a person but was not recognized by Microsoft Face API because of the mask.

Microsoft computer overcomes both above scenarios was able to recognize both images as shown in Figure (4) and Figure (5).

Figure (4) shows the same image shown in Figure (2) but Microsoft Computer Vision service was able to recognize the image.

Figure (5) shows the same image shown in Figure (3) but Microsoft Computer Vision service was able to recognize the image.

The vision service performs the following tasks:

- Iterates through all images and send them to the computer vision API.
- Receives the response and check the number of persons inside the image by check special keywords like 'person', 'man', 'women', 'boy', and 'girl'.

- Update the image table and set HasPeople to true if there are two or more people in the images.
- If one or zero person on the image, we set HasPeople to false. The system doesn't send images that has only one person since it is not applicable from the social distance perspective.

The computer vision service main job is to help with reducing the load on the human computation. If search engine run in a production environment or if we have thousands of images that we want to analyze, then the computer vision service can help us with filtering out the images that are not applicable to the issue that we are trying to solve. The images that is labeled by the workers can be used to train the model to have a better prediction to filter out more applicable images.

3.3.3 Image Hashing

We crawl images from news agencies and perform a random google image search. There is a chance that the same images will get send to MTurk more than one time since news agencies show the same images for popular events. This duplication have the following consequences:

- Cost money because we need to pay more workers
- Cost time because we need to wait more to receive responses
- Lead to incorrect result where the same image will be labeled differently and can be displayed in two different categories at the same time. The same image can be displayed with the images that adhere to the social distance rules, and it can also displayed in the images that don't adhere to the social distance rules.

To calculate the image hash, we perform the following tasks:

- Download each image and convert it to an array of bytes
- Calculate the object hash [*] and store the hash in the database
- Load the hash of all images into in-memory dictionary

The process of using the image hashing in the system is listed below:

- The system read images from the database and filter out the images that are not applicable for human computation from the social distance perspective
- Iterate through all images, calculate the image hash, and check it in the in-memory dictionary of hashes to see if the image exists or not.
- If the image doesn't not exist, then the system will send it to MTurk API.
- If the image exists, then the image record will get updated, and the previously identical image information will be fetched and update the current image with the MTurk response.

3.3.4 MTurk Service

The MTurk service is responsible for sending requests and receiving responses to and from Amazon Mechanical API. The service performs the following tasks:

- Reads images from the database
- Perform validation on the image
- Check if the images has been identified by the computer vision service as applicable
- Check if the image hash is unique, and the image was not sent before.
- Read the MTurk request object that has a configurable image URL.
- Replace the URL with the current image URL
- Send the image to MTurk API.
- Update the database with the HIT ID and the Request URL.
- Periodically check the MTurk assignments and fetch the workers responses
- Update the database with the responses

The image database allows two iterations of request/response pair and can save two Hit Ids and two responses. In our study, we run the system into two iterations to validate the responses, and we design the database to respond to these two iterations. We will talk more about than in the result section.
The image request is shown in Figure (6).

We have added a text to explain the big picture [11] of our task to educate the worker about the work that we are doing and the impact that they can contribute to. We hope that the explanation can help with motivate the people to answer the question more accurately.

Figure (6) MTurk request form.

3.3.5 Challenges

On challenge with our approach is that the images that we crawled from the news agencies are outdated or not enough to perform the labeling or analysis. The news articles contain only one image or very few about grocery stores or workplace. We addressed this issue by augmenting our image dataset by searching most recent images from Google Images to make the image dataset up to date.

An interested research challenges that we faced with the social distance image labeling is that many pictures has two persons who seems from one family either shopping or running outside. People from the same family or live together are most likely both have coronavirus or don't have coronavirus. The computer vision will specify that the image has persons and the workers will specify the distance as less than 6 feet. While it is hard to know if two persons are from one family or not, but this is an interesting research problem that can be studied. One idea will be if both are taking or smiling that might give an indication that they are from one family.

3.3.6 Technologies

We used different technologies to build our system.

Programming Language: We used C# language to build the backend and HTML, CSS, JavaScript to build the front end.

Database: We used SQL Server database to store our data. We built different tables for our system and created primary key, foreign keys, and indices to maintain our database integrity. We also used Microsoft Excel to export and import the data

Web Development: We used ASP.NET Core to build our front end with razor pages.
Design Patterns: We used Model-View-Controller in our system to abstract different application layer and make the code more modular.

Amazon Mechanical Turk (MTurk): we used MTurk SDK to build the MTurk service to send requests and receive responses.

Microsoft Computer Vision API: we used Microsoft Computer vision to analyze images to find the number of persons in the image.

4 Evaluation

Our main goal of the system is to show images to the users that adhere to social distance rules, that don't adhere to the rules or not applicable. In order to achieve the goal, we are evaluating all components of the system as listed below:

- We expect the system to crawl images from different sources about coronavirus. We have successfully achieved this goal, by crawling images from news agency in different categories related to coronavirus and we have also augmented our image dataset with other images from google images, and the actual images were indeed about coronavirus.
- We expect the system to send images to a computer vision service to filter out images that has zero or one person. We have successfully achieved this goal, and we are sending images to Microsoft Vision service and we received good results from the service
- We expect the system to send images to crowdsource workers to use human computation to label images. We have successfully achieved this goal by sending images to Mechanical Turk, and we have hired workers to perform the labeling for us, and we have received their responses.
- We expect the system to send only unique images and not send duplicate images to MTurk. We achieved this goal by implement-

ing image hashing and calculate the image hash before sending it to the MTurk API.

- We expect the system to show images in different categories related to social distances. We have successfully achieved this goal by building a web interface that shows images in three categories: not applicable images for images that have no people or a single person, images that adhere to social distancing rule that has at least two persons with minimum distance of 6 feet between them, images that don't adhere to social distance that have at least two persons and the distance between them is less than 6 feet.

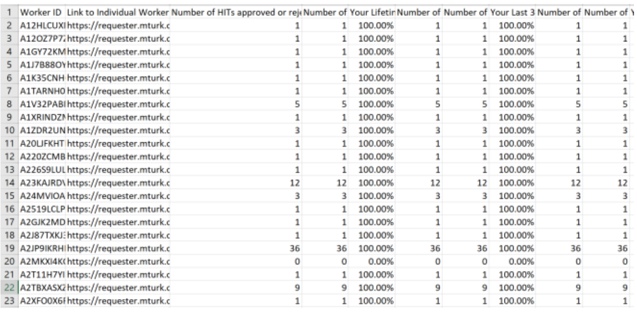
5 Results

Images: we have selected 120 images for our study, and these images are from the sources shown in Table (1):

Total images	Images from news agencies	Images from Google Images
120	70	50

Table (1) Shows the number of images crawled and manually collected.

MTurk Workers: we hired 54 workers and most of them worked on one task, but there are few of them who worked on multiple tasks. The maximum tasks per worker was 36. Figure (7) shows some the workers report.



1	Worker ID	Link to Individual Worker	Number of HITs approved or reje	Number of	Your Lifetin	Number of	Number of	Your Last 3	Number of	Number of Y
2	A12HLCUXI	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1
3	A12OZ7P7	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1
4	A1GY72KM	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1
5	A1J7B88OY	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1
6	A1K35CNH	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1
7	A1TARNHO	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1
8	A1V32PABI	https://requester.mturk.c	5	5	100.00%	5	5	100.00%	5	5
9	A1XRINDZN	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1
10	A1ZDR2UN	https://requester.mturk.c	3	3	100.00%	3	3	100.00%	3	3
11	A20LIFKHT	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1
12	A220ZCMB	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1
13	A226S9LUL	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1
14	A23KAJRD\	https://requester.mturk.c	12	12	100.00%	12	12	100.00%	12	12
15	A24MVIOA	https://requester.mturk.c	3	3	100.00%	3	3	100.00%	3	3
16	A2519LCLP	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1
17	A2GJK2MD	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1
18	A2J87TXKJ	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1
19	A2JP9IKRHI	https://requester.mturk.c	36	36	100.00%	36	36	100.00%	36	36
20	A2MKXI4K(	https://requester.mturk.c	0	0	0.00%	0	0	0.00%	0	0
21	A2T11H7YI	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1
22	A2TBXASX2	https://requester.mturk.c	9	9	100.00%	9	9	100.00%	9	9
23	A2XFO0X6I	https://requester.mturk.c	1	1	100.00%	1	1	100.00%	1	1

Figure (7) MTurk worker responses.

Social Distance Results: The final number of images in each category are shown in Table (2):

Not applicable images	Adhere to social distance rules	Don't confirm with social distance rules
52	20	48

Table (2) shows the number of images in each category.

We have 41% of images that adhere to social distance rules and 59% of images that don't adhere to social distance rules.

Discrepancies: we run to iteration of the system to send the 120 images for labeling two times to validate the response of the workers. We found discrepancies in 26 images. We define discrepancies as one response below 6 feet and the other response more than 6 feet.

Gold Standard Data: we checked all the responses from the two iterations. We were interested in the images that have discrepancies. We manually checked these images and looked at them to determine which worker response is correct. We didn't check manually any image that has both responses below 6 feet or both responses that are equal or above 6 feet.

Figure (8) shows the images that are not applicable. We can see the first and second images have people, but they are far away and very small which made both the human computation and the AI failed to recognize.

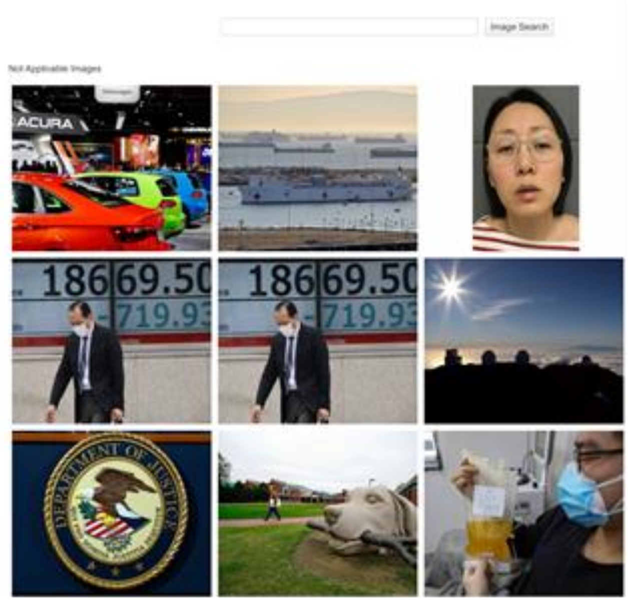


Figure (8) Not applicable images.

Figure (9) shows the images that adhere to the social distance rules.

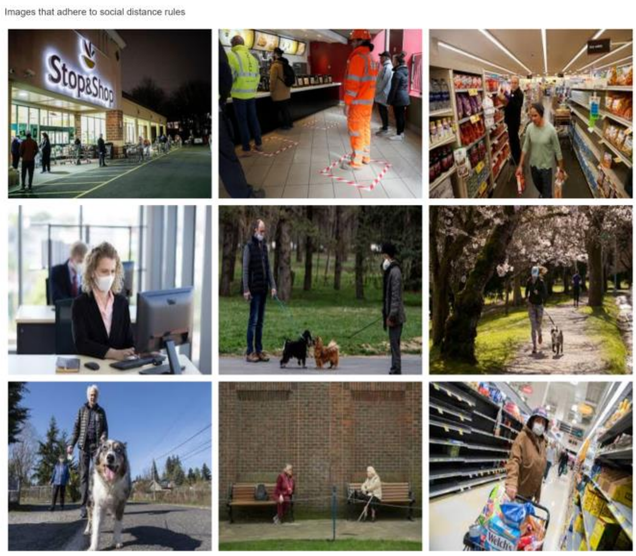


Figure (9) Images that adhere to social distance rules

Figure (10) shows the images that don’t adhere to the social distance rules. We can see in the fist picture that people are wearing special suites that protect them from being infected but both AI and Human didn’t flag it. We asked the crowdsource worker to specify the distance, but we were expecting that workers will specify that this image is safe, but that didn’t happen.

Figure (10) Images that don’t adhere to social distance rules.

Figure (11) shows the result of a text search when the user search for a query related to coronavirus.

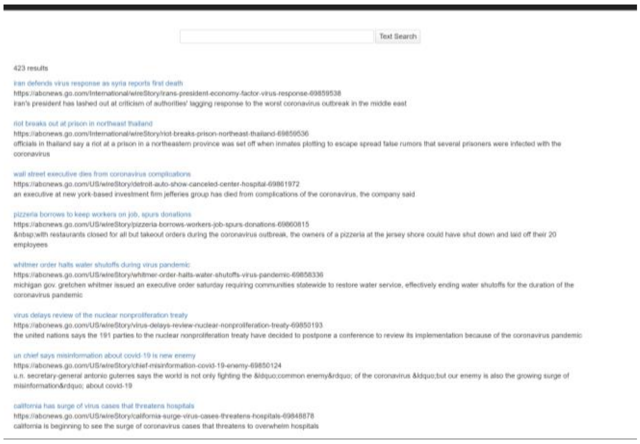


Figure (11) text search results for coronavirus query.

6 Broader Impacts

The system crawl news articles and images from popular news agencies and doesn’t deal with personal information, but the images retrieved have people faces and we are assuming that the news agencies take privacy into their consideration or they will otherwise be sued. We have also augmented the images dataset with images from Google Images and these images are public too. Therefore, we are not taking any privacy measures into our system design.

The system can be used to help people to protect themselves from being infected by the coronavirus while leaving their houses for essential business to go to groceries stores, doctor offices, and workplace. The system can also be used by companies to enforce guidelines to help protecting their employees and customers.

The system can also be implemented in a different setting to provide guidelines about any large impact crises as an awareness tool to provide guidelines to help people stay safe.

Lastly, the system can be used for evil if workers intentionally label images incorrectly but that can be easily discovered by the search engine user.

References

[1] Donna Vakharia and Matthew Lease. 2015. Beyond Mechanical Turk: An Analysis of Paid Crowd Work Platforms. In iConference, 17.

[2] World Health Organization

[3] Danna Gurari, Suyog Jain, Margrit Betke, and Kristen Grauman. 2016. Pull the Plug? Predicting If Computers or Humans Should Segment Images. 382–391

[4] Jean Y Song, Stephan J Lemmer, Michael Xieyang Liu, Shiyan Yan, Juho Kim, Jason J Corso, and Walter S Lasecki. 2019 b. Popup: reconstructing 3D video using particle filtering to aggregate crowd responses. In Proceedings of the 24th International Conference on Intelligent User Interfaces (IUI'19). ACM, New York, NY, USA, 558--569. https://doi.org/10.1145/3301275.3302305

[5] K. Hara, V. Le, and J. Froehlich. Combining crowdsourcing and google street view to identify street-level accessibility problems. In CHI, 2013.

[6] Gierad Laput, Walter S Lasecki, Jason Wiese, Robert Xiao, Jeffrey P Bigham, and Chris Harrison. 2015. Zensors: Adaptive, rapidly deployable, human-intelligent sensor feeds. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems. ACM, 1935--1944.

[7] Ma, H. and Chandrasekar, R. and Quirk, C. and Gupta, A. Page hunt: improving search engines using human computation games. In Proc. of SIGIR 2009, 746—747.

[8] https://www.newyorker.com/culture/culture-desk/mapping-the-sidewalks-of-new-york-city-for-social-distancing.

[9] Modern Information Retrieval: the concepts and technology behind search (2nd Ed)

[10] Text-Image Based Search: Optimization of document raking with deep visual and text features, Kosaka Sato and Mohannad Al Ameedi 2017

[11] Nathan Hahn, Joseph Chang, Ji Eun Kim, and Aniket Kittur. 2016. The Knowledge Accelerator: Big Picture Thinking in Small Pieces. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI ’16), 2258–2270