NETS 213 Final Project Report

Basic project information

Name of your project
Pickture Perfect

Name of your teammates
Ben Robinov, Eva Killenberg, Kieran Halloran, Shivani Komma, Tiffany Tsang

Give a one sentence description of your project. Please use the name of the project in your description.
Pickture Perfect uses crowd workers to select the best quality pictures from large sets of images.

Logo for your project. Create a PNG file, save it in your github repo, give its URL.
https://github.com/tiffanytsang/pickture-perfect/blob/master/pickture_perfect_logo.png

What problem does it solve?
Pickture Perfect solves the problem of sifting through a large set of photos, say from a vacation, to find the best images.

What similar projects exist?
Google Photos as well as several Android and iOS applications perform this function through AI, but Pickture Perfect is the only crowd-powered solution.

What type of project is it?
Human computation algorithm

What was the main focus of your team's effort
Conducting an in-depth analysis of data

How does your project work? Describe each of the steps involved in your project. What parts are done by the crowd, and what parts will be done automatically.
We created 6 sample sets of images of various sizes, all from trips we have taken. There are two sets of 50 pictures, two sets of 220 pictures, one set of 360 pictures, and one set of 440 pictures. Ten percent of the pictures in each of these sets were

purposefully doctored as negative quality control images. To do this, we inverted the colors of the image, making it an obviously bad image.

We posted these image sets to Mechanical Turk in two different HIT designs meant to narrow down the set to only the good photos. The first HIT design simply asked workers if the image was good or bad. The second HIT design asked workers to check off descriptions that applied to the image. There were positive qualities such as “Good composition: main subject is framed as focus of the image,” and negative ones such as “Discolored, oversaturated, undersaturated, and/or poor lighting.” Each image would be rated by 3 workers.

For the Good/Bad HIT, we performed quality control by eliminating all submissions by the worker if they marked a negative quality control image as “Good.” We then aggregated these results by ranking the images based on the fraction of times the image was ranked good.

For the Checkbox HIT, we performed quality control by removing the results of all crowd workers who either: (a) selected more than 4 checkboxes for a single image, meaning that they selected good and bad checkboxes for the same characteristic, (b) checked a box for a characteristic that is contingent on the presence of people AND checked a box that said there are no people in the image, or (c) selected more positive characteristic checkboxes than negative characteristic checkboxes for a negative QC image. We then aggregated these results by ranking the images according to this score formula: $\frac{good - bad + 1}{bad + 1}$. We used this calculation as it weighted bad characteristics more heavily than good ones and gave higher ratings to pictures with people.

From both HITs, we took the top 20% of the rankings to pass on to the next round. There was a single HIT design in this round, which asked you to select the top three images out of a set of ten, where one of the ten was a negative quality control image. We randomized the ten images that appeared, but made sure that each image was shown three times, as was done in the first round of HITs. The quality control component in this round disqualified any submissions that selected the bad photo as one of the top three. We then ranked the images in order of the number of times it was selected in the good subset, and took the top 25% of this list as the final set of best photos (5% of the entire photoset). This HIT was performed on the results from both the Checkbox and the Good/Bad HITs for each photoset.

We then analyzed the results. We compared the final sets generated by the crowd workers to our own favorite images as a “Gold Standard,” we analyzed how the cost of

the HITs scaled with the size of the photoset, and we compared cost and efficacy between the two different HIT designs in the first round.

Provide a link to your final presentation video. Give the full path to your Vimeo video, and the password, if it is not public.

The Crowd

Who are the members of your crowd?
Mechanical Turk Workers

For your final project, did you simulate the crowd or run a real experiment?
Real crowd

If the crowd was real, how did you recruit participants?
We recruited participants by paying them on Mechanical Turk.

How many unique participants did you have?
554

Incentives

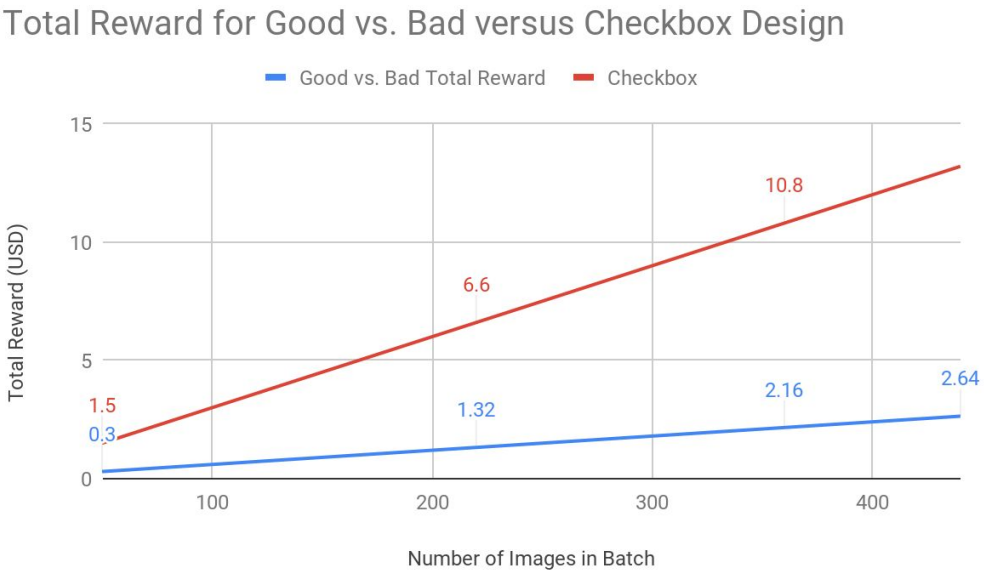
What motivation does the crowd have for participating in your project?
We motivated the crowd by paying them through Amazon's Mechanical Turk platform.

How do you incentivize the crowd to participate? Please write 1-3 paragraphs giving the specifics of how you incentivize the crowd. If your crowd is simulated, then what would you need to do to incentivize a real crowd?
We incentivize the crowd with micropayments in the form of money via Mechanical Turk. For the good vs. bad design, we pay Mechanical Turkers 2 cents to identify 10 images as either good or bad. All of these 10 images are on one screen. For the checkbox design, we pay Mechanical Turkers 1 cent to attribute at least 1 of 8 qualities to one image. For the subset design, we pay Mechanical Turkers 1 cent to pick the best 3 photos out of 10. All of these 10 images are on one screen.

Did you perform any analysis comparing different incentives?
Yes

If you compared different incentives, what analysis did you perform? If you have a graph analyzing incentives, include a link to a PNG file of the graph here.

When comparing the HIT designs, as a worker, it is more economically beneficial to do the checkbox HITs, as each image pays 1 cent, while for the other HITs, each image analyzed pays less. We generated a graph that compares the total reward to the worker across the good vs. bad HIT design and checkbox HIT design. The scaling up section will discuss this topic further.



What the crowd gives you

What does the crowd provide for you?
The crowd provides answers to our HIT designs. These answers are the best images in a set of vacation photos.

Is this something that could be automated?
Yes, to an extent.

If it could be automated, say how. If it is difficult or impossible to automate, say why.
AI could be trained to recognize images with certain qualities. However, it is a lot harder to train an AI to recognize a “good photo.” We think humans have a unique perspective to pick out the nuances of what makes a photo good, which is why crowdsourcing is a good solution to this problem.

Did you train a machine learning component from what the crowd gave you?

No

Did you create a user interface for the crowd workers? Answer yes even if it's something simple like a HTML form on CrowdFlower.
Yes, two MTurk HITs and a Qualtrics survey.

If yes, please give the URL to a screenshot of the crowd-facing user interface. Save the screenshot as a PNG file, and put it in your github repo. Include the full path to your image (prefix with https://github.com/). You can include multiple screenshots, one per line.
https://github.com/tiffanytsang/pickture-perfect/tree/master/data/Checkbox%20Screenshot.png
https://github.com/tiffanytsang/pickture-perfect/tree/master/data/GoodBad%20Screenshot.png
https://github.com/tiffanytsang/pickture-perfect/tree/master/data/Qualtrics%20Screenshot.png

Describe your crowd-facing user interface. This can be a short caption for the screenshot. Alternately, if you put a lot of effort into the interface design, you can give a longer explanation of what you did.
For the checkbox HIT, we first displayed detailed directions on how to decipher the positive and negative qualities of images. Then the worker would click "Proceed," and be presented with a single image, along with multiple checkboxes, as shown in the first screenshot.
For the good/bad HIT, we first show the same directions as the checkbox HIT, and after the worker clicks "Proceed," they are presented with 10 images, each with a radio button to mark it as good or bad. After marking all 10 images, they may submit.
We created a Qualtrics survey for our subset HIT, as this functionality wasn't supported as well in MTurk. After being presented with the same directions as in the other HITs, the crowdworker is presented with 10 images and they are tasked with selecting the 3 best images.

Skills

Do your crowd workers need specialized skills?
No

What sort of skills do they need?

They need to have good vision and a very basic understanding of factors that constitute high image quality.

Do the skills of individual workers vary widely?
No

If skills vary widely, what factors cause one person to be better than another?
We think that the average worker will be reasonably capable of analyzing the quality of an image. If a worker was vision impaired, they may not be able to do this, or if they are a professional photographer, they may be able to do this particularly well. However, these are rarities among Mechanical Turk workers, so it will not make a significant difference.

Did you analyze the skills of the crowd?
No

If you analyzed skills, what analysis did you perform? How did you analyze their skills? What questions did you investigate? Did you look at the quality of their results? Did you analyze the time it took individuals to complete the task? What conclusions did you reach?
N/A

Do you have a graph analyzing skills? If you have a graph analyzing skills, include a link to a PNG file of the graph here.
No

Quality Control

Is the quality of what the crowd gives you a concern?
Yes. Because we are recruiting Mechanical Turk workers, whose only incentive is a very small amount of money, we implemented quality control measures in all of our HITs. It would be easy for Mechanical Turk workers to randomly click answers on our HITs before submitting.

How do you ensure the quality of the crowd provides?
We implemented various quality control standards that we will describe in depth in the next question. In short, we sought to ensure that crowd workers were not randomly clicking buttons and checking boxes, as this behavior would skew our results.

If quality if a concern, then what did you do for quality control? If it is not a concern, then what about the design of your system obviates the need for explicit QC? This answer should be substantial (several paragraphs long).

One quality control mechanism was to make obviously bad quality photos by inverting the colors. Here is how we used this technique in each HIT:

> Good vs Bad: If the worker classified a negative QC image as a "good" image, their results were excluded from the analysis.
> Checkboxes: If a worker ever selected more positive characteristic checkboxes than negative characteristic checkboxes for a negative QC image, their results were excluded from the analysis.
> Subset: If the worker selected a negative QC image to be in the top subset of images in that set, their results were excluded from the analysis.

Additionally, for the checkbox HIT, we excluded the results of all crowd workers who either: (a) selected more than 4 checkboxes for a single image, meaning that they selected good and bad checkboxes for the same characteristic, (b) checked a box for a characteristic that is contingent on the presence of people AND checked a box that said there are no people in the image— both of these behaviors indicate carelessness and/or random clicking.

Did you analyze the quality of what you got back? For instance, did you compare the quality of results against a gold standard? Did you compare different QC strategies?

As our HITs were quite simple, we sought to uphold a very high standard of quality through a strict disqualification policy for careless work. However, we were pleasantly surprised by the small amount of low quality work received. For each HIT, only a handful of Turkers had their results excluded by our QC measures. As such, we were satisfied with the work quality and our quality control procedures. Thus, we didn't see the need to experiment with other QC strategies.

What analysis did you perform on quality? What questions did you investigate? What conclusions did you reach?

We analyzed our QC procedures, specifically the percentage of workers that were disqualified by our QC module. We looked at possible correlations between the size of the photoset and the disqualified percentage, as well as comparing the percentages

disqualified in the Checkbox HIT vs. the Good/Bad HIT. Lastly, we investigated the breakdown of the reasons for disqualification in the Checkbox HIT.

The first graph shows the percentages of workers whose results were disqualified in our Checkbox HIT. The percentages range from just under 2% to around 15%, with the mean at 7.06% and the standard deviation at 5.03%. We also found that the correlation between the size of the photoset and the percentage of workers disqualified is 0.091142.

The second figure shows a breakdown of the reasons why workers' results were disqualified in our Checkbox HIT. Almost half of these disqualifications stemmed from workers indicating that a negative QC image (i.e. an image with obviously distorted color and saturation) had proper coloration and/or saturation. The other two reasons each comprised roughly one quarter of the disqualifications. The first was if a worker checked more than 4 checkboxes. As there were 8 checkboxes, 4 pairs of good/bad characterististics, checking more than 4 would indicate contradiction and therefore carelessness. The second was if a worker checked a checkbox that was contingent on the presence of people in the image and also selected the button that indicated there are no people in said image. This, once again indicates contradiction and therefore carelessness.

The last figure shows percentages of workers whose results were disqualified in our Good/Bad HIT. Here, the percentages range from 0% to 25%, with the mean, very similar to the Checkbox HIT, at 7.8632% and the standard deviation at 9.7117%. In this case, the correlation between the size of the photoset and the percentage of workers disqualified is actually negative, at -0.22005. Further, this figure and the corresponding Checkbox graph look quite similar, which is confirmed by their correlation: 0.83352.

In summary, we found minimal to no correlation between the size of the photoset and the percentage of workers disqualified. Further, we found high correlation between the two HIT designs for percentage disqualified by photoset. All of this evidence indicates that the contents of a photoset is a more influential factor than its size with regards to the percentage of disqualified workers. Also, when looking at reasons for disqualification, our results were basically as expected: the two blatantly contradictory behaviors each comprised roughly a quarter of the total.The remainder of the disqualifications were workers marking a negative quality control image as good. We anticipated fewer disqualifications from first two "careless" behaviors and thus a larger percentage for our negative QC images but we stand by our QC module and attempts to reject random clicking.

We subsequently performed the same analysis on the results from the second round (subset) HITs. In this HIT design, workers were presented with 10 photos, one of which was a negative QC image, and were asked to select their top 3 images. We disqualified the work of any worker who placed a negative QC image in the top 3. We expected there to be fewer disqualifications since this cutoff is less strict than the ones imposed on the initial round of HITs. However, for this HIT design we could not think of a stricter way of implementing negative QC. The results met our expectations, as out of 105 unique workers on the subset HIT, only 1 was disqualified for selecting a negative QC image.

Do you have a graph analyzing quality? If you have a graph analyzing quality, include a link to a PNG file of the graph here.
https://github.com/tiffanytsang/pickture-perfect/blob/master/data/QCAnalysisCheckboxPcts.png
https://github.com/tiffanytsang/pickture-perfect/blob/master/data/QCReasonsCheckbox.png
https://github.com/tiffanytsang/pickture-perfect/blob/master/data/goodbadDQPcts.png

Aggregation

How do you aggregate the results from the crowd?
We aggregated the results from the crowd by processing the responses of the MTurk HITs for the first round (checkbox and good/bad designs), and the results of the Qualtrics survey for the second round (subset design). In the first round, our goal was to get the top 20% of images from the original set to pass into the second round HIT. We did this using 2 different methodologies. For good vs. bad, this consisted of ranking the images by what percentage of qualified workers classified it as “Good.” For checkbox, we used a slightly more complicated method. We first classified each checkbox as either “good” or “bad.” For instance, the checkbox titled, “Good composition: main subject is framed as focus of the image” was labeled “good,” while the checkbox titled, “Main subject is blurry” was labelled “bad.” After counting the total number of good and bad votes for each image after removing unqualified workers, we scored each image using the following formula: $\frac{good - bad + 1}{bad + 1}$.

For the second round HIT, our goal was to produce the final set of images, which would be 5% of the size of the original set. Again, we first disqualified any workers according

to our QC criteria described previously. We then sorted the images by the number of times each was selected as one of the top 3 images of its group of 10, then taking the top 25% of images from this ranking.

Did you analyze the aggregated results?
Yes

What analysis did you perform on the aggregated results? What questions did you investigate? Did you compare aggregated responses against individual responses? What conclusions did you reach?
We first analyzed the overlap between the top 20% of images produced by the two first-round HIT designs. We classified “overlap” as the percentage of photos shared by the top 20% of results from each HIT design. We found that the overlap ranged from approximately 30 to 40 percent for all the photosets.

For the aggregated results from the subset HIT, we analyzed how much the outputted images (top 5% of the entire photoset) produced overlapped, depending on whether the input came from the results of the good/bad HIT design or checkbox HIT design. We found that the overlap decreased even more, as the overlap for each photoset ranged from 0% to approximately 22%.

Do you have a graph analyzing the aggregated results? If you have a graph analyzing the aggregated results, include a link to a PNG file of the graph here.
https://raw.githubusercontent.com/tiffanytsang/pickture-perfect/master/data/OverlapAnalysis.png
https://raw.githubusercontent.com/tiffanytsang/pickture-perfect/master/data/round2_overlap_graph.png

Did you create a user interface for the end users to see the aggregated results? If yes, please give the URL to a screenshot of the user interface for the end user. Save the screenshot as a PNG file, and put it in your github repo. Include the full path to your image (prefix with https://github.com/). You can include multiple screenshots, one per line.
No

Describe what your end user sees in this interface. This can be a short caption for the screenshot. Alternately, if you put a lot of effort into the interface design, you can give a longer explanation of what you did.
N/A

Scaling Up

What is the scale of the problem that you are trying to solve?

The scale of the problem depends on the number of people who would like to use this service. Currently, the project is more of a proof of concept and research study, but in theory, Pickture Perfect would help curate the best photos in a large set of photos. A greater number of crowd workers would be needed if outsourcing best image selection is a high-demand service. We could potentially partner with photos storage sites such as Flickr, Google Photos, or Dropbox, which have millions of users. Alternatively, we could run as an independent service, in which case demand is dependent on the amount of marketing.

Would your project benefit if you could get contributions from thousands of people?
Yes, depending on the demand for this image selection service.

If it would benefit from a huge crowd, how would it benefit?
We could process more vacation photo sets, as more people would be available to rate the quality of each image. We could also assign more than 3 people per HIT instance to ensure more significant results. For instance, if a greater number of people identify a particular photo as a good image, then it is more likely that the particular image is actually good.

What challenges would scaling to a large crowd introduce?
If we choose to assign more than 3 people per HIT instance to ensure more significant results, we need to determine the number of people to assign per HIT. The answer to this question depends on the demand for the image selection service. If there are lots of input images, we would like to decrease the number of crowd workers working on a particular HIT instance to save costs. It would also be interesting to study whether the quality of results actually increases if more than 3 workers analyze the same image. Other than cost, scaling to a large crowd would not pose significant challenges. Creating negative images for quality control is an automated process, and the aggregation analysis would not greatly differ.

Did you perform an analysis about how to scale up your project? For instance, a cost analysis?
Yes

What analysis did you perform on the scaling up?
We analyzed the total time to complete a batch, the total cost to requesters, the average time spent per HIT assignment, and the hourly wage. Across these analyses, we varied the size of the input image set as well as the HIT design for the first round. The first round consists of the good vs. bad and checkbox designs. For the first round, we analyzed the results for image sets of 3 sizes: 50 images, 220 images, and approximately 400 images. For the second round, we also performed the same analysis for the subset HITs. We analyzed the results for the top 20% of photos from round 1 in addition to negative control images, which means we have results for image sets of 3 sizes: 11 images, 50 images, and approximately 90 images.

What questions did you investigate? What conclusions did you reach?
For the first round of HITs, to filter out the top 20% of photos, we were interested in whether the good vs. bad design or the checkbox design would be more effective, in terms of cost, time, and quality of work. The quality of work will be discussed in the project analysis section. As for cost and time analysis, we analyzed differences in cost and time across multiple sizes of input image sets as well as the two different HIT designs. As a point of comparison, we also overlay the figures with data from the round 2 subset HITs.

Figure 1: This figure compares the total time it took for MTurkers to complete the batches across the 3 HIT designs. This was calculated by finding the average time spent per HIT, then scaling it based on the number of images per batch. We observed that it took 13.52 times longer for the good vs. bad HITs to finish as compared to the checkbox HITs. As this system scales, it would be more time-effective to use the checkbox HIT design. Meanwhile, the subset HITs took about 2.01 times longer to finish as compared to the checkbox HITs.

Figure 2: This figure compares the total cost for requesters to upload a batch to MTurk across the 3 HIT designs. The difference between the good vs. bad batches and checkbox batches is that the cost for checkbox batches quickly increased as the input image set size got larger. In fact, it is five times more expensive to get an image in the checkbox HIT design analyzed by a crowd worker, since it costs $0.002 to analyze an image in the good vs. bad HIT design and $0.01 to analyze an image in the checkbox HIT design. For future work, it might be more cost-efficient to utilize the good vs. bad design. Meanwhile, the subset HITs cost about the same as the good vs. bad HITs.

Figure 3: This figure compares the average time spent per assignment across the 3 HIT designs. MTurkers spent about 13.52 times longer on the good vs. bad HITs than on the

checkbox HITs. This makes sense because we placed 10 images per good vs. bad HIT, but only 1 image per checkbox HIT. When we analyze the average time spent per image, we see that MTurkers spent about 1.35 times longer on the good vs. bad HIT design, rather than the checkbox design even though the checkbox HIT takes more effort. This is an interesting outcome, but again suggests that it would be more time-effective to utilize the checkbox design. Meanwhile, the MTurkers spent about 2.01 longer on the subset HITs than on the checkbox HITs. When we take into consideration that the MTurkers evaluate 10 images per subset HIT, we conclude that it takes about 4.98 times less time to evaluate an image for the subset HITs as compared to the checkbox HIT.

Figure 4: This figure compares the hourly wage across the two HIT designs in round 1. The hourly wage is computed by taking the cost per image (reward to MTurkers) and dividing it by the average time spent per image. We observed that workers on the checkbox HIT earn an hourly wage that is about 6.76 times more than workers on the good vs. bad HIT. This seems unfair, since the good vs. bad HITs took longer per image and overall. This suggests that we should either lower the cost of the checkbox HIT in future work, or only use the good vs. bad HIT design. If we choose to lower the cost of the checkbox HIT, we could accomplish this by inserting more images per HIT. The hourly wage for the subset HIT falls between the wages for the good vs. bad design and checkbox design. Perhaps we could look to the subset HIT hourly wage to find a good compromise on an hourly wage for the round 1 HITs.

In summary, the good vs. bad HIT design was more cost-effective for the requester, but more time intensive for the crowd worker. The good vs. bad HIT design paid less per image and also took longer for workers to complete. In terms of cost, as the project scales, it would be beneficial for Pickture Perfect to perform the good vs. bad analysis rather than the checkbox analysis. If time is of importance, then the checkbox HIT might be a better option. However, this difference in time is perhaps because we placed one image per HIT rather than multiple. In terms of scalability, we think that cost is more important than time, and so as the system scales, we would probably use the good vs. bad design.

As for the subset HITs, the total time to complete the subset HITs is about double that of the checkbox HITs. The subset design costs about the same as the good vs. bad design. The average time spent per subset HIT is greater than the average time spent per checkbox HIT, while the average time per image is less. The hourly wage for subset fell in between that of the good vs. bad design and checkbox design. Overall, the costs and time data for the subset HIT fell between the two extremes of our good vs. bad

design and checkbox design. Therefore, the decision of whether or not to include the subset HIT for extra filtering is outsourced to the project analysis section. If it results in better quality of filtered images, it would be worth implementing the extra filtering step, as it costs about the same as the cheaper good vs. bad design and takes about the same time as the shorter checkbox design. Otherwise, there is no reason to double the cost and time to get top images that are of equivalent or perhaps worse quality as compared to the results from round 1 HITs.

Do you have a graph analyzing scaling? If you have a graph analyzing scaling, include a link to a PNG file of the graph here.

Figure 1 (trendline in lighter color):

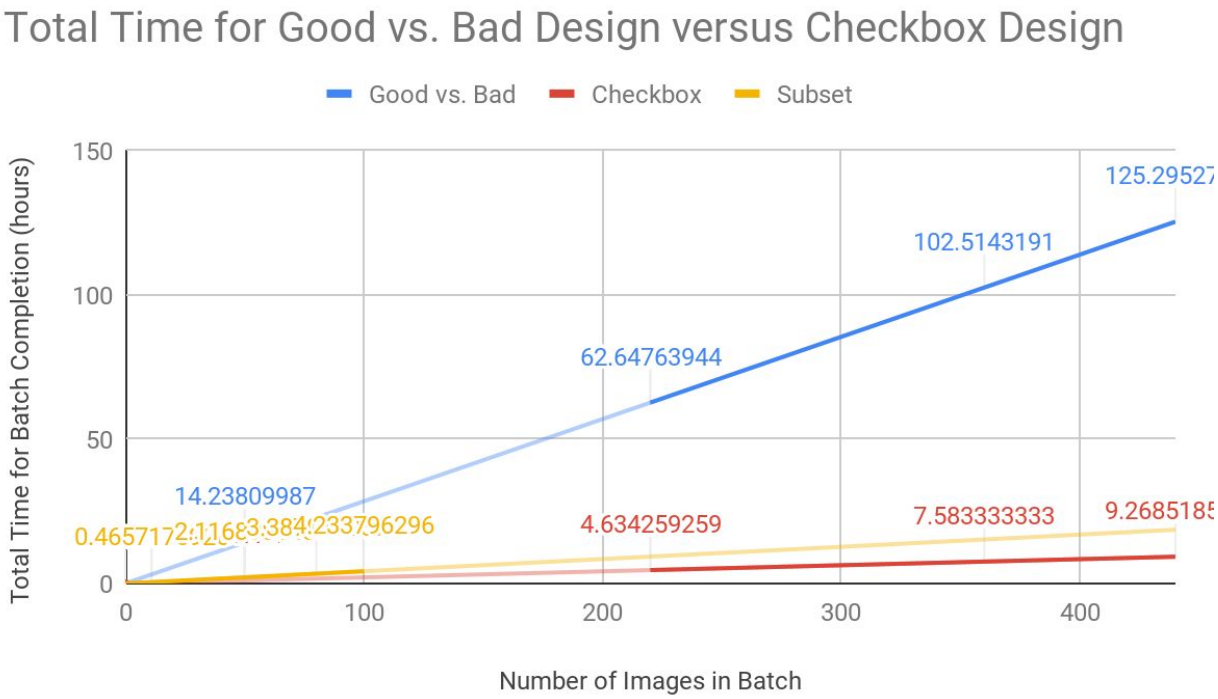


Figure 2 (trendline in lighter color):

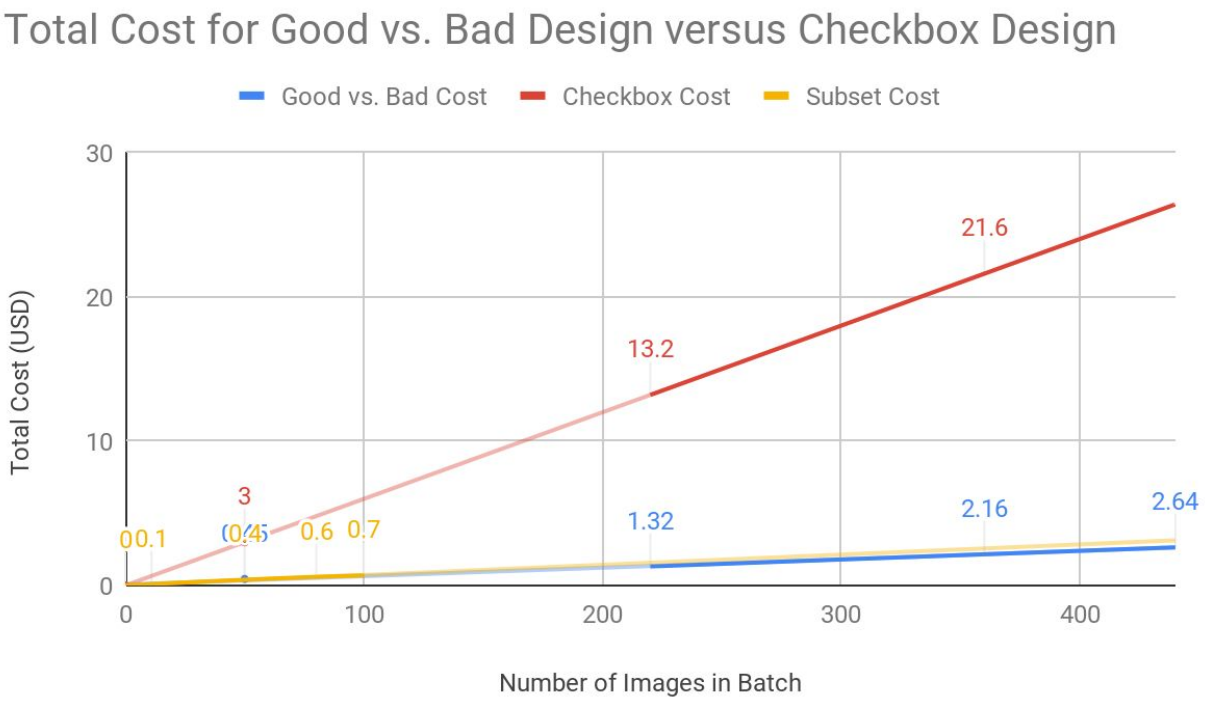


Figure 3:

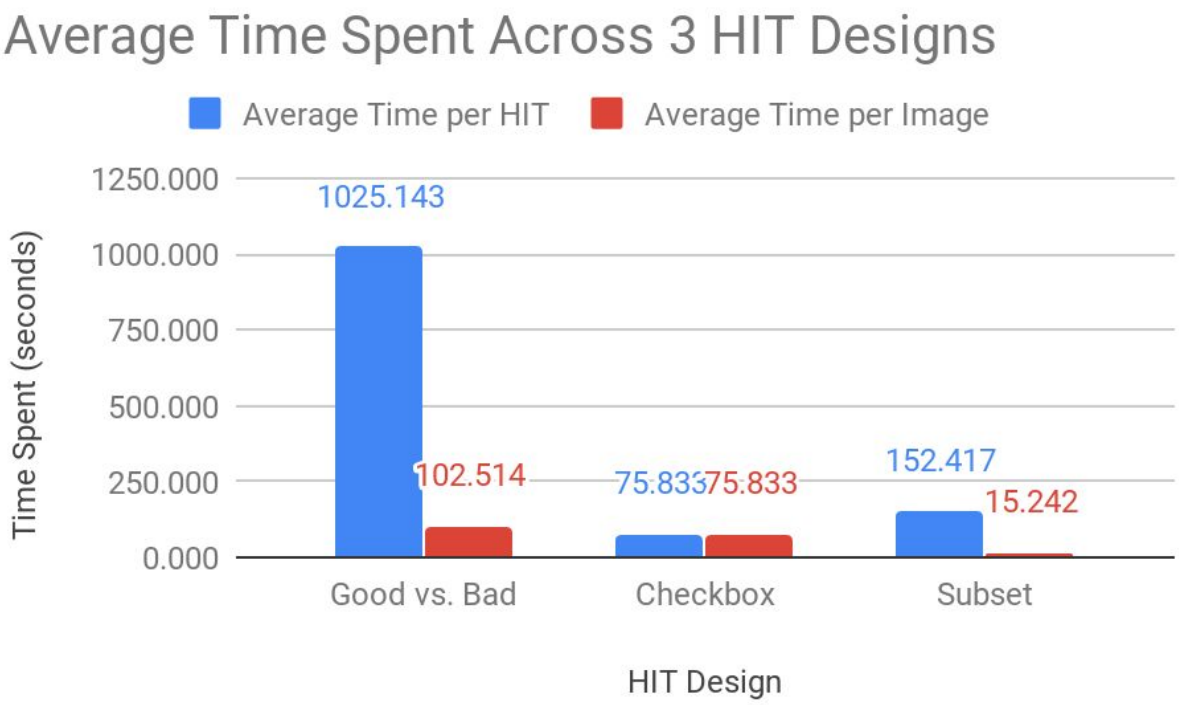
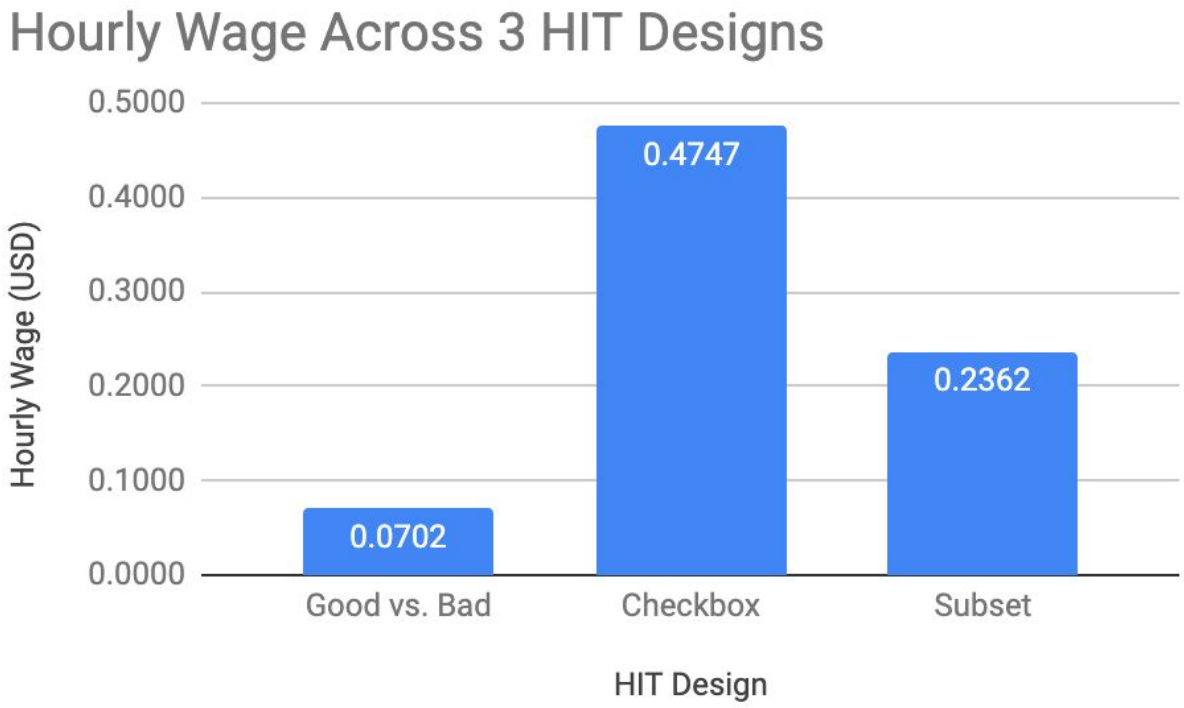


Figure 4:



Project Analysis

Did your project work? How do you know? Analyze some results, discuss some positive outcomes of your project.

We think that our project worked quite well, but still has some limitations.

To see how well our project worked, for each image set, we manually choose our favorite 5% of images, and then compared these to the top 5% of images selected by the good/bad and checkbox processes (see graph). Often times, there was overlap between the images we chose and the images produced by the crowd. Neither the good/bad design nor the checkbox design consistently had more overlap with our images. These results demonstrate that our project is able to effectively identify top images from a very large photo set.

We also manually looked through the top images, and verified that the top images chosen by the algorithm were all high quality images. Before putting the images through the subset HIT, the results from the checkbox HIT seem to provide a better ranking of the images than from the Good/Bad HIT. After putting the images through

the subset HIT, the results from both the checkbox and Good/Bad hit seem to be of similar quality.

After manually looking through the top images ranked by the initial Checkbox HIT, it is not clear to us that the quality of the top images significantly improved after putting them through the round of Subset HITs. As such, it might be not be necessary to put the images through the additional round of HITs and select the top 5% based just off of the initial round of HITs. As discussed in the scaling up analysis, it is quite expensive to do the Checkbox HIT, which is further reason not to incur more cost from a second round of Subset HITs.

The two main advantages of the Checkbox HIT design over the Good/Bad and Subset designs are that it allows us to specify criteria for a good or bad image, and also provides a more granular ranking than the other HIT designs. For instance, our design has a slight bias towards photos with people who are looking at the camera, since we had a checkbox for pictures with people in them that asked whether or not the subjects were looking at the camera. Additionally, the formula that we use to calculate the score for an image gives more distinction between images. For instance, there are only a few possible percentages from which we can rank the results of the Good/Bad HIT, and there are only 4 possible scores (0, 1, 2 and 3) that an image can receive from the Subset HIT. The Checkbox HIT, on the other hand, gives a range of scores with many more possible values, thanks to the aggregation formula we used.

Arguably, the biggest issue with Pickture Perfect is that it does not permit the crowd to consider how a group of images would be amalgamated into a top collection. This is something that a person would definitely consider when selecting the best photos manually. Nonetheless, we can confidently say that Pickture Perfect accomplishes our underlying goal of selecting some of the best individual images from a large set of vacation photos, thereby saving users the time and effort to sort through these photos themselves.

Do you have a graph analyzing your project? If you have a graph analyzing your project, include a link to a PNG file of the graph here.
https://github.com/tiffanytsang/pickture-perfect/blob/master/data/our_best_overlap_graph.png

What were the biggest challenges that you had to deal with?
We had to change our direction a few times, which was quite frustrating as we had put a lot of time and effort in work which was then rendered obsolete. Additionally, creating

the HITs in MTurk was a bit difficult and unintuitive. We had a hard time figuring out which form elements to use to best meet our needs for the HITs. Because we were unfamiliar with HTML and Javascript, the learning curve for designing HITs in Mechanical Turk was quite steep. We also saw that we were only able to show one screen by designing a HIT on mTurk. We wanted to show instructions on one page and have the images on another page. We also wanted to be able to upload the entire photo set to one HIT and then display a select few of them for the user to choose from. All of these features were available on Qualtrics and not the default Mechanical Turk platform so we decided to create the HIT in a Qualtrics survey and then provide a link to it in the HIT we posted on mTurk.

Where there major changes between what you originally proposed and your final product? If so, what changed between your original plan and your final product?
Yes, there were major changes between our initial proposal and final product. Originally, our project was more of a scientific study. We planned to use three different HIT designs to complete the task of choosing the best image out of a set of 10 similar images. We created three HIT designs: X vs. Y (comparing two images side by side), subset (selecting the best images out of a set), and ranking (rating each image 1 to 10). Each design would generate the best photo or set of photos from an album. The photos we initially used were very similar, taken seconds apart. While they were not as similar as a picture taken with Burst on an iPhone, they were mostly from the same moment in the same setting. We wanted to determine the best photo beforehand and then compare this image with the results of the three HIT designs. The output would tell us which HIT design was the most accurate method of finding the best photo(s).

We received feedback, however, that this project was too similar to what Google Photos and iPhone Live Photos does when selecting the best frame or image from a set of similar photos. A more pressing problem could be returning from vacation and having hundreds or even thousands of photos to condense. This would require Turkers to subjectively assess which photos are the best

We adapted our original design, updating two HIT designs (X vs. Y, subsets) and varying sizes of photosets to analyze the HIT designs and how cost scales with the size of the set. We modified our photo sets to be larger and include a greater variety of pictures as one has after going on vacation. The details of the project are described at the beginning of this report.

What are some limitations of your product? If yours is an engineering-heavy project, what would you need to overcome in order to scale (cost/incentives/QC…)? If yours

was a scientific study, what are some sources of error that may have been introduced by your method.
A limitation is the way we collected data on the photos. The subset design only had 3 votes/image because we had 3 workers look at each photo, which does not have as much granularity as other HITs. To have more detail, it would be ideal if more workers could look at each photo. However, this is expensive so it would have to be within the constraints of a budget. Also, as previously discussed, the lack of global knowledge and perspective for each worker can limit the diversity in the final set of top photos.

Did your results deviate from what you would expect from previous work or what you learned in the class? If your results deviated, why might that be?
In class, the most similar algorithm we learned was using Quicksort to identify the cutest kitten pictures. It was a helpful start to see how comparing images could lead to the finding the best photo. We wanted to first select and rank good images in the first round and then further narrow down the images in the second round. This was not very similar to Quicksort-ing the kittens so we could not compare our results.

Technical Challenges

Did your project require a substantial technical component? Did it require substantial software engineering? Did you need to learn a new language or API?
Our project required us to learn Qualtrics for the second round HITs. Mechanical Turk only allowed us to display one screen and required knowledge of HTML and Javascript. Through free Penn accounts, we could take advantage of Qualtrics' built-in survey features like importing image sets as choices, choice randomization, evenly present counts, validation codes, and separate screens. We describe importing image sets in the next question and the other features below.

Once the images were available, workers had to select the best three out of every ten images. With choice randomization, we were able to specify that a subset of 10 randomly selected images should be displayed during each HIT. This also allowed us to place the negative images at the bottom of each photoset and trust that they would appear in a random number and at a random place within each subset of 10. Using the "evenly present elements" function, we required that the survey show each image the same number of times. Because we had three workers look at each image, each image would appear in a survey three times. The evenly present elements function allowed us to put all the images from a set into one survey but have each image display in a set of

10. This way we wouldn't have to create many different links; instead, one link would correspond with one image set.

Additionally, we added a randomly generated number as embedded data. At the end of the survey, users would take the number displayed and enter it back into Mechanical Turk to receive payment.

We were able to create this HIT across multiple screens in Qualtrics. Screens included collecting the mTurk ID, instructions, selecting the top 3 images, and then the validation code. The variety of features built into Qualtrics allowed us great flexibility in survey design. Given the benefits, it was worth learning how to manipulate Qualtrics for our second round HIT.

If project required a substantial technical component, describe the largest technical challenge you faced.
The largest technical challenge was importing the large image sets into Qualtrics. Our smallest set was 50 images and our largest was 440 images. Because the worker had to select their favorite three out of every ten images during the second round, each option needed to display an image. Given that we had 1,340 images and two HIT designs (doubling the number of images that needed to be seen), we needed to find an efficient way to display each photo.

How did you overcome this challenge? What new tools or skills were required? Help us understand the amount of work that was required.
We began by manually uploading each photo as an option in a multiple choice question. This method became quite tedious, especially as the number of photos grew. Additionally, it was difficult to see if we had accidentally uploaded the same photo twice because we could not see all the images at once. We were only able to see two images at a time rather than all the urls.

For our next attempt, we created labels like "img1" that were associated with a url like "https://s3.amazonaws.com/pickture-perfect/singapore/img_1.jpg" in the embedded data section. We then populated the question choices with the labels. While this was faster than uploading each individual image and allowed us to compare urls a bit more quickly, it was still tedious when we had to do that 2,680 times (across all six sets and two HIT designs).

Finally, we found an "Edit Multiple" option that allowed us to specify the name of the answer choice (e.g. Image 1) and displayed the image based on an HTML line. For

example, this line “img1 <img src='https://s3.amazonaws.com/pickture-perfect/singapore/img_20.jpg' style='max-height:400px; padding: 10px;'></img>” would display the label “img1” to the user and image 20 from the Singapore set. We then listed all the images in Excel in one column (i.e. img1, img2, img3), the beginning HTML code in one, the image URL in the next, and the ending HTML code in the last. We then concatenated the columns and saved the resulting strings as a CSV. The URLs were the top 10% of the previous HIT design results. We then copy pasted the column and pasted it into Qualtrics. This populated the answer choices and drastically reduced the time it took to upload all the images into one survey.

Do you have any screenshots or flow diagrams to illustrate the technical component you described? If so, include a link to a PNG file of the graph here.

Other info (optional)

Is there anything else you’d like to say about your project? If you have additional information about your project that didn’t fit into the above questions, put it here.
This project was a really interesting way to get hands-on experience with crowdsourcing and data analysis. We are very proud of our results and how far we’ve come since our original ideas!