How to Publish HITs on Amazon Mechanical Turk

Jianxiong Xiao

Abstract: This is a practical introduction to teach how to use Amazon Mechanical Turk, using the diversity experiments in my NIPS 2013 paper "Learning Deep Features for Scene Recognition using Places Database" as an example.

In general, there are 3 ways to use Turk.
1. template: the easiest way and probably most reliable way
2. command line tool: not easy, and not reliable
3. API: not easy but reliable. Useful for sophisticated control and

If you want to use 3.API, I have written a Matlab API interface that you can use
http://vision.princeton.edu/pvt/MatlabTurkTool/
see the demo.m file

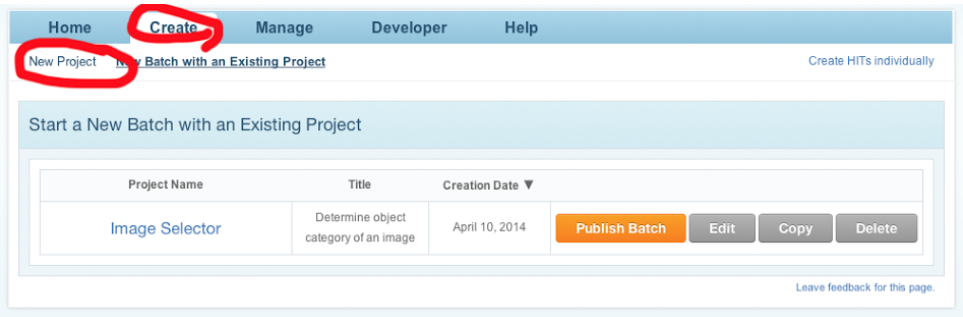
For the places experiment, we use 1.template. The following text describes how to use template only.

Here is the the source code for our template
http://vision.princeton.edu/projects/2014/places/12AFC.html
right click on the webpage to see the source code.
The code starts is like this:
<html>
<body>
...
</body>
</html>

That means that we put all html, javascript, and css into the body, because we are going to copy the body content text to Amazon.

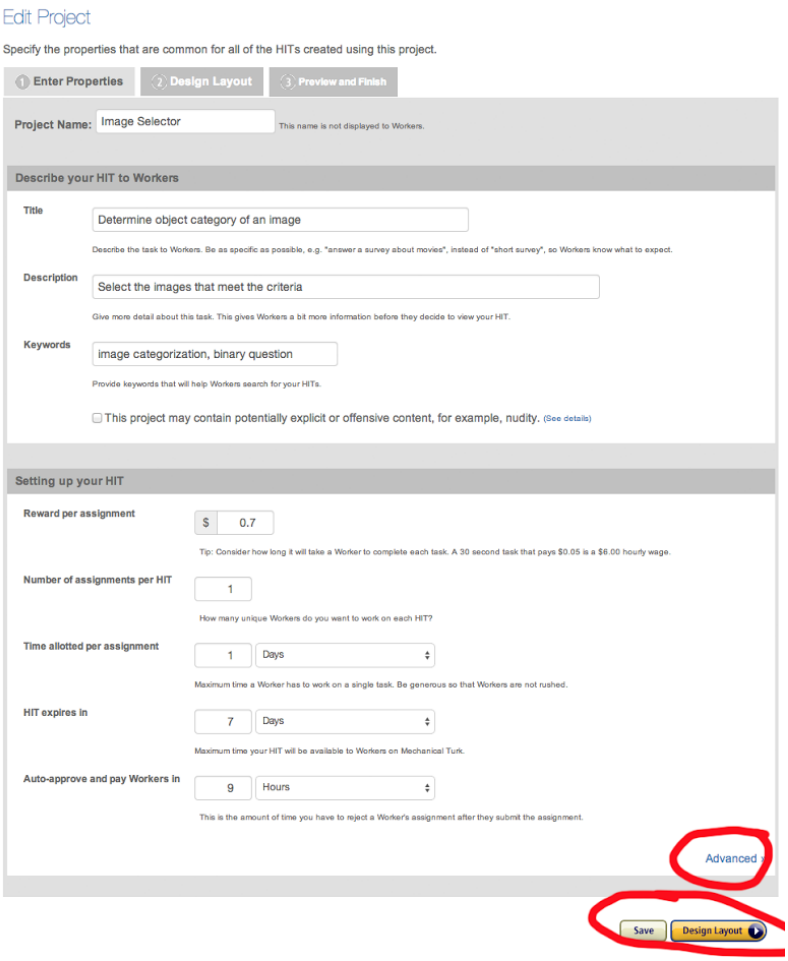
Now, login to the website
http://requester.mturk.com
or
http://requestersandbox.mturk.com

Create a new template



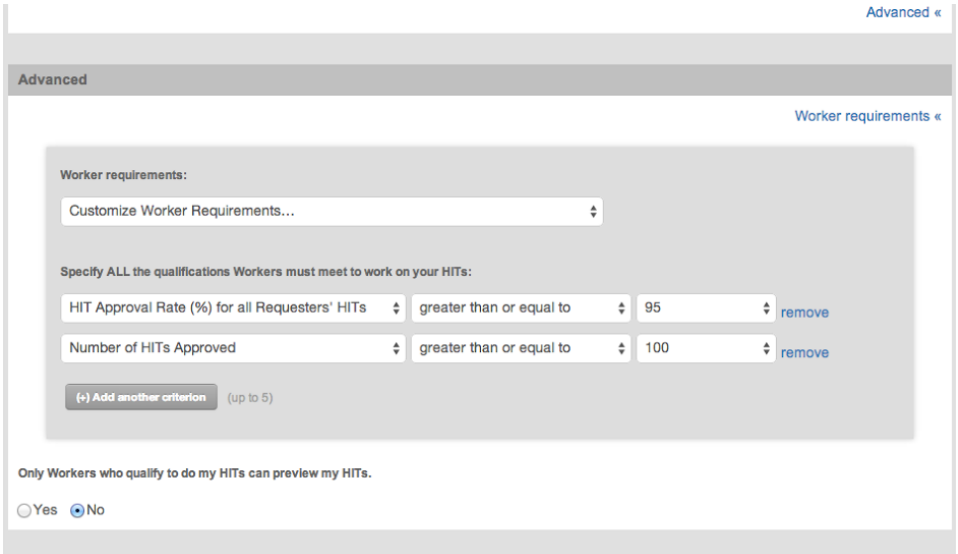
You can choose any default template provided by Amazon, because we are going to remove the content and replace it by our code completely.

Fill in the form:

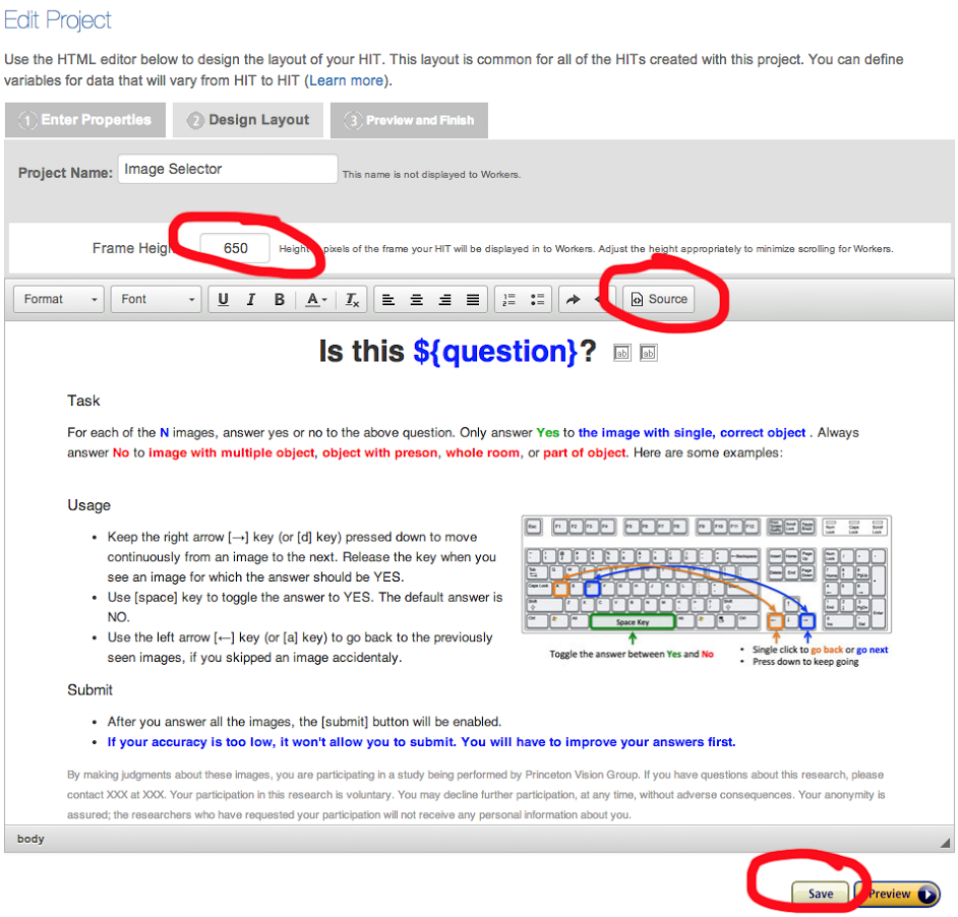


Usually, we set an automatic approval time since we pay everyone.
It is arguable how long to set this time.

Click on [Advanced]
Here you get to set the criteria for workers. Amazon sets it to use Masters by default, which means we will pay more money to Amazon and the jobs will be done by workers picked by Amazon as masters. Usually we don't do it. So we will remove that criteria, and Amazon will have a warning.



Now we can start to design the layout

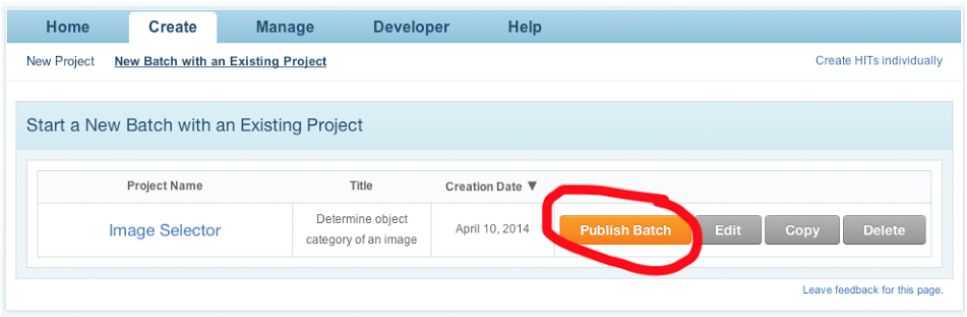


First, there are two modes for editing. We only use the source code mode. Click the [Source] Button to switch to source mode, and then, delete whatever text is there. Replace the text there by our webpage without <html><body></body></html>. Only the content. Don't copy <html><body></body></html>. Now you can click and switch back to the WYSIWYG view. But don't change anything here, because the WYSIWYG editor may do something stupid that destroy our code.

Now, you need to set the height of the webpage. The HIT will be display as an iframe. So we need to set the height of the iframe. Usually we set a value that is big enough so that the turker doesn't need to be scrolling to finish the task. Otherwise, there will be two scrolling bar, one for the whole webpage, one for the iframe inside the webpage, which are very annoying and confusing. So, it is very important to set the height of the iframe.

Now you can [save] the template. Note that you can save only if it is not in the source mode (which is stupid, but Amazon decides to do this).

After we have a template, we just need to upload a CSV file for the list of images and we are done. Choose the template you want, and select [Publish Batch] to choose a .csv file.



A .csv file is just a table.
In the first line, it will list the table header, which match to the ${XXXXX} in the webpage. Amazon will automatically replace the ${XXXXX} in our template by the actual content in the .csv file.
Here is an example file
http://vision.princeton.edu/projects/2014/places/input_first100.csv

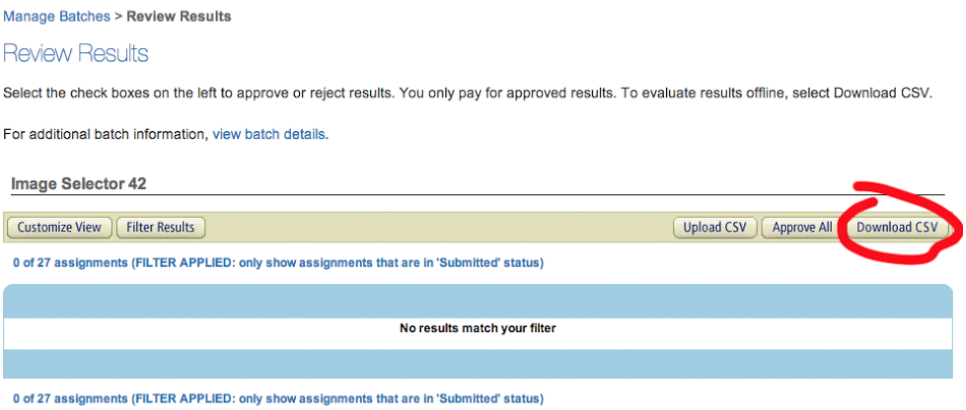
For example, our template has
<img class="pair" onclick="selectChange(0)" id="img0" src="${imgname0}">
<img class="pair" onclick="selectChange(1)" id="img1" src="${imgname1}">
<img class="pair" onclick="selectChange(2)" id="img2" src="${imgname2}">
Then, the first line of the .csv file is
imgname0, imgname1, imgname2

The following lines are the image list. Each line is one HIT.
We can have as many columns as we need from the source code.

Usually, we generate the .csv file by Matlab, e.g.
http://vision.princeton.edu/projects/2014/places/genInput.m

You can open the .csv file by Excel to view the table. But don't save it. Excel may introduce some strange format and change the file so much that Amazon cannot recognize.

After we publish the HITs and the workers finish the jobs, we can download the result as a .csv file



Then, you can use my Matlab function to read the .csv file into Matlab and do the analysis accordingly.

http://vision.princeton.edu/code.html#readCSVfromMTurk

For example, here is the code that I used to extract the result

http://vision.princeton.edu/projects/2014/places/extract_result.m

For the programming of HIT, it is basically just a html form.
But we want to create some hidden input to put the data.
So when the tuker hits the submit button, the result will be passed on to Amazon to generate the .csv output file.

One special thing for Amazon is that we can tell if the HIT is accepted by
if (gup('assignmentId') == "ASSIGNMENT_ID_NOT_AVAILABLE")
If it is not, we don't allow the worker to work or submit.

And that is it!