



Actua's AI Activities Series

Activity 5

Hand Commands: Training Image Classification Models

Hand Commands: Training Image Classification Models

If you're accessing this activity directly, did you know there are eight other activities in this series up on our website? If you find yourself unfamiliar with any of the AI concepts and terminology introduced in these activities, please refer to our [AI Glossary](#). These activities also follow a space exploration narrative when done in order. It is recommended to complete the activities in order but they can also be done on their own.

You and your group-mates are astronauts and scientists aboard the Actua Orbital Station. Unfortunately, your station just got bombarded by magnetic rays and your electronics have begun to shut down! The only one who can save you is the orbital station's AI, DANN. DANN stands for Dedicated Actua Neural Network, and it's gone a little loopy. Brush up on your technical skills, learn about AI, and save yourself and your crewmates!

DANN's visual core is almost complete, thanks to your hard work in "[Sort Things Out: Exploring Image Classification](#)". But we aren't done yet! Before the magnetic storm, DANN was a state-of-the-art AI system that you could interact with using your voice. But during the shut-down, DANN's audio core was knocked offline. The problem is, it doesn't look like we can access the audio core without DANN's help first. Our mission specialists think that you might be able to use DANN's visual core to communicate, but the visual core isn't working either! We need to train an image classification model to recognize our hand shapes as commands, and then make sure they're refined and ready to go in "[What Machines See: Digging into Machine Vision](#)"!



Activity Summary

In this activity, participants will train an AI model to complete an image classification task. They will use Google's Teachable Machine platform to classify images of hand gestures that can be used to control a computer program. Participants will create datasets to use for training the AI model and in doing so, will learn about how AI programs understand the information that they're presented with.

Developed by Actua, 2022.

Delivery Environment	Activity Duration	Intended Audience
Classroom with computer access	30-45 minutes	Grades 9-12 (Ages 13-18)

Achievement Goals

Learning Goals

Learning goals are statements referring to the understanding, knowledge, skills or application participants acquire during the activity. **Following this activity, participants will:**

- **Identify and describe** the process of image classification
- **Investigate** the strengths and weaknesses of programs in specific situations
- **Develop and test** a machine learning model



Logistics (Timing, Group Size, Materials)

Section Title	Time	Group Size	Materials
Activity 1: Training your AI model	20 minutes	<i>Individual/ Small groups</i>	Group Size <ul style="list-style-type: none">• Device with internet access• Functioning webcam• Teachable Machine
Activity 2: Using your AI model	15 minutes	<i>Individual/ Small groups</i>	Group Size <ul style="list-style-type: none">• Device with internet access• Functioning webcam• Teachable Machine• DANN Image Classifier
Reflection & Debrief	5 minutes	<i>Whole group</i>	

Safety Considerations

Safety considerations have been provided below to support safety during this activity, however they are not necessarily comprehensive. It is important that you review the activity and your delivery environment to determine any additional safety considerations that you should be implementing for the delivery of these activities.



Online Safety

Some components of this activity require the use of devices connected to the internet.

- Facilitators should review the provided videos and read/explore provided websites and materials to determine if they are suitable for their participants.
- Where applicable, facilitators should remind participants to stay on task and only use links provided within this activity.

Activity Procedure

Our mission specialists think that you might be able to use DANN's visual core to communicate, but the visual core isn't working. We need to train an image classification model to recognize our hand shapes so that we can get access to the audio core and reinitialize it. Reinitializing one of DANN's cores is a fairly straightforward process, but the steps have to be completed in a certain order. Most of the steps use specific commands:

1. DISENGAGE core safety interlocks.
2. DISMOUNT core from station operating system.
3. SHUTDOWN core.
4. REINITIALIZE core.

The commands within each step are executed using the keyword in all-caps. For example, to complete the first step, we need to send a "DISENGAGE" command. If a core is successfully reinitialized, it should be automatically engaged by the station's operating system. You'll need a way to give these four commands, in order, and we can't use DANN's audio systems to do it.



Section 1: Training your AI Model

The two pre-trained models encountered in “Sort things out: Exploring Image Classification”, while impressive in their own way and potentially useful elsewhere, don’t seem like a good fit for what we need. They can identify many different objects with varying degrees of success, but they aren’t currently set up for recognizing the hand shapes that you will be using as commands. It’s looking like you might need to train your own model to help you reinitialize DANN’s audio core. Luckily, the mission specialists included instructions to help you through the process. They read:

IN CASE OF EMERGENCY, A CLASSIFICATION MODEL CAN BE TRAINED USING TEACHABLE MACHINE.

1. LOAD TEACHABLE MACHINE
2. DEFINE CLASSES
3. CREATE TRAINING DATA
4. TRAIN MODEL
5. TEST MODEL

AFTER MODEL IS TRAINED AND TESTED, IT CAN BE CONNECTED TO AN INPUT CORE.

Load Teachable Machine

Teachable Machine can be accessed online here:

<https://teachablemachine.withgoogle.com/train/image>.

This link will bring you to Teachable Machine’s image classification model. Teachable Machine has two other models: a body pose-based classification model and an audio-based classification model. We will explore these in later activities. For now, you should focus on training Teachable Machine’s image classification model to recognize the hand shapes that we’ll be using to communicate with DANN.



Define classes

Defining classes is a key part of a classification task: you need to tell your model what you are looking to identify (e.g. “I think this image is in Class A, and not Class B or C”). As humans, we can rely on a number of sensory inputs (as well as our own memories) to help us complete classification tasks. An image classification AI model, on the other hand, only has the data from the images that you’ve shown it, to make its guesses as to which class an image might fit into. It can only make informed guesses.

Remembering back to our initial briefing, you should recall that there are four commands, one for each step in the core reinitialization process. We need the image classifier to recognize each one:

- DISENGAGE
- DISMOUNT
- SHUTDOWN
- REINITIALIZE

These will be the four classes that you will need your classifier to recognize, but you should also add a class for when the model doesn’t recognize any of these classes: an UNKNOWN label.

1. In Teachable Machine, rename any existing classes (Class 1 and Class 2, by default) to the names of the commands.
2. Add the remaining classes, by clicking on “+ Add a class” below your existing classes, for each of the remaining commands and “UNKNOWN”.
3. When you are done, you should have five boxes, one for each class. They should be to the left of the middle box entitled “Training”.



Create training data

For each of the four commands, you will need to come up with a unique hand shape.

Facilitator note: Ensure that all shapes chosen are appropriate and not offensive. If in doubt, provide predetermined options for participants to choose from.

You will be teaching your AI model to recognize these shapes through a process called training. Your image classification model learns by looking at visual examples of each class, so that it can begin to tell the difference between them. Rather than relying on existing images of your chosen hand gestures, which may be difficult to find, you will create the training images using Teachable Machine's built-in webcam function. The webcam captures images that the model can then use for training. The mission specialists included some instructions on how to proceed with this phase:

1. Define and/or agree upon a hand shape for each of the four commands.
2. Sketch these shapes in the space provided.
 - a. Note: these shapes don't have to convey any actual meaning (e.g., your shape for DISENGAGE doesn't have to somehow communicate the idea of disengaging). The only requirement is that the shapes need to be distinct enough that the program can tell them apart easily.
 - b. In Teachable Machine, for each command class:
 - c. Click on the "Webcam" button in the class box. You may be asked for permission to use your webcam. After training, unless you specifically save your Teachable Machine project, the images gathered will be discarded.
 - d. Record images of your hand shape. You have two options for recording:
 - i. Use "Hold to Record" to gather images yourself or have another person press "Hold to Record" for you. You should record for 3 to 5 seconds.
 - ii. Click on the gear icon (⚙️) to access the recording settings and turn "Hold to Record" off. This will enable a timed recording mode that you can use to record your images for a fixed duration after a small delay, suggested to be 3 to 5 seconds.



- e. If you have group mates or a friend nearby, repeat the recording process with them also making the hand shapes.
3. For your UNKNOWN class:
 - a. Repeat the recording process but for hand shapes that do not resemble your command classes. The purpose of this class is to provide your model with a non-command class guess option. Make sure you make lots of different hand shapes for this label, as it is meant to be the catch-all for any other gestures!

Train Model

Now that you've created examples of what you want your AI model to recognize, you can have your model examine the images in each of your classes and begin distinguishing between them. To do this:

1. Click on the "Train Model" button in the box labelled "Training".
2. Wait for training to complete. After a short while, below the training button, you should see a timer counting up and a number out of 50. Your training is complete once that number reaches 50 out of 50, but, depending on the number of training images you have, this might take a few moments.
3. When training is complete, the "Preview" box should have an "Output" section which displays your trained model's classification of your webcam image and its confidence in its classification.

Test Model

You need to check if your model's been trained well enough for you to use it to communicate with DANN's visual core. This means that your model should reliably detect each hand shape that you trained it on and classify everything else as UNKNOWN. The mission specialists have included some questions to consider for evaluating if your model is functioning well:

1. Does your model accurately classify, for each of your command classes, your specified hand shapes...
 - a. ...when made by you or other group members used in the training data?



3. You should be able to click on “Upload my model”. Click “Upload my model”.
4. It may take a few moments (10-15 seconds in good Internet conditions) for your model to be uploaded, but once it’s complete, you should now have a link under “Your shareable link:”. This link should be in the following format:
<https://teachablemachine.withgoogle.com/models/xx-xxxxxx/>
 - a. This is the URL to your trained model. We can use this to connect your model to DANN’s visual core!
5. Activate the window below and paste the URL for your Teachable Machine model where the window prompts you to. This will load your AI model into DANN’s visual core.
6. This step will also describe steps to exporting your model. The sketch will use the provided model and require a hand shape to be successfully detected for an amount of time before accepting it as a command. Confidence threshold will be adjustable.
 - a. [DANN Image Classifier](#)
7. Once your model is loaded, you can adjust the confidence threshold. Any images below your threshold will be ignored by the operating system. The timing threshold indicates how long you have to hold the shape for it to be recognized.
8. Using your hand shapes, you must enter the commands to re-initialize the visual core (in order).
 - a. The core has a failsafe to help limit incorrect inputs, so you must hold each hand shape until the system has confirmed that your input has been registered.



Reflection & Debrief

As you enter the last command, you feel a wave of relief as the screen displays the text “REINITIALIZATION IN PROGRESS”. Time slows as you wait for the reinitialization process to conclude, but eventually, a new notification appears: “AUDIO CORE ONLINE”.

While you're eager to move on to working on the newly reinitialized audio core, first we must think back to the questions that we used to evaluate our trained models:

1. Does your model accurately classify, for each of your command classes, your specified hand shapes...
 - a. ...when made by you or other group members used in the training data?
 - b. ...when made by other people who were not part of the training data?
 - c. ...when applied in a different environment from where the training data was generated (e.g. a different part of the room, a different background for the webcam, different levels of light)?
2. **{ Individually, reflect on / As a whole class, discuss }** the appropriateness of these questions for confirming that your model is adequately functional, specifically:
 - a. What is the difference between these three questions? Do you think the differences are important? If so, why? If not, why not?
 - b. What might the answer to each question tell us about how well our model is trained?
3. Other questions to reflect on:
 - a. Do you think shapes or gestures are a good way to interact with a computer system? Why? Why not?
 - b. Not including voice control (which will be addressed in a subsequent activity), what other ways could you improve the interactions with the station's systems and DANN's AI?
 - c. How could visual recognition of hand gestures be applied to real world problems? Where could it make things easier?
 - d. Can you think of any AI used in the real world that have similar functions to Teachable Machine? How are they useful?



Delivery Adaptations

How might you adapt the time, space, materials, group sizes, or instructions to make this activity more approachable or more challenging? **Modifications** are ways to make the activity more accessible, **extensions** are ways to make the activity last longer or more challenging.

Modifications

- If a participant is struggling with the activity, focus on one gesture to begin with. See if the program can distinguish between two possibilities: The gesture is being made, or it isn't. Create data for those two options, and train the model.
- In a virtual environment, each participant is able to run through the activity on their own. Ensure that they have access to the proper links.
- If participants wish to combine training data, Teachable Machine allows you to zip input images and add them to a google drive. From there, another computer can install them directly into Teachable Machine.

Extensions

- Expand the training data for your image classification model. How many different gestures can you train it to recognize? Provide each label with more training data. How does that affect the quality of the model? Are there any significant changes? Does the average confidence go up?
- Have someone else test your trained model. Does it recognize their commands? What could you do to improve that recognition?



References & Gratitude

Google's Teachable Machine helped make this activity possible. It can be found at <https://teachablemachine.withgoogle.com/train/image>.

Terms of Use

Prior to using this activity or parts thereof, you agree and understand that:

- It is your responsibility to review all aspects of this activity and ensure safety measures are in place for the protection of all involved parties.
- Any safety precautions contained in the "Safety Considerations" section of this write-up are not intended as a complete list or to replace your own safety review process.
- Actua shall not be responsible or liable for any damage that may occur due to your use of this content.
- This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. For more information, please see <https://creativecommons.org/licenses/by-nc-sa/4.0/>.
- You may adapt the content for your program (remix, transform, and build upon the material), providing appropriate credit to Actua and indicating if changes were made. No sharing of content with third parties without written permission from Actua.

About Actua

Actua is Canada's leading science, technology, engineering and mathematics (STEM) youth outreach network, representing a growing network of over 40 universities and colleges across the country. Each year 350,000 young Canadians in over 500 communities nationwide are inspired through hands-on educational workshops, camps and community outreach initiatives. Actua focuses on the engagement of underrepresented youth through specialized programs for Indigenous youth, girls and young women, at-risk youth and youth living in Northern and remote communities. For more information, please visit us online at www.actua.ca and on social media: [Twitter](#), [Facebook](#), [Instagram](#) and [YouTube](#)!



Appendices

Appendix A: Career & Mentor Connections

- Machine learning researcher
- Sign Language Interpreter
- Computer Programmer
- Data Analyst

