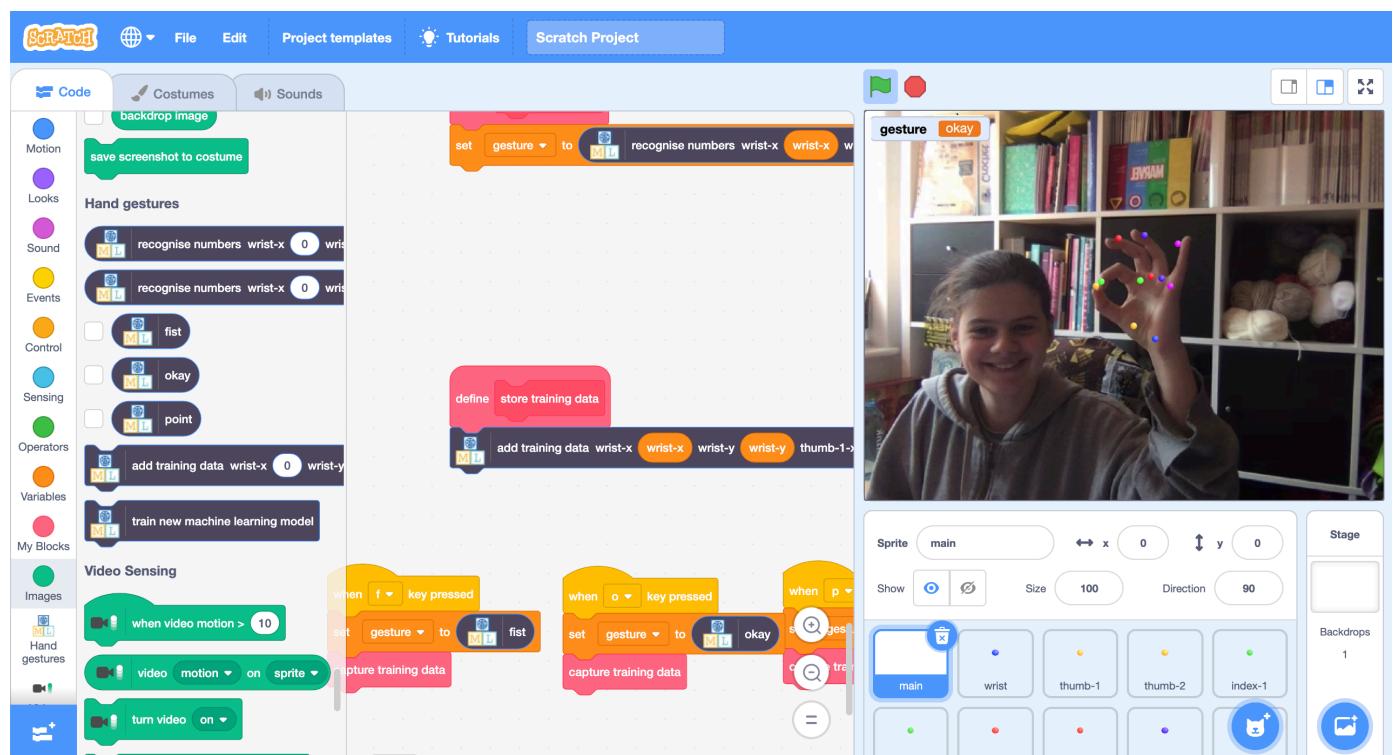




Hand gestures

In this project you will train the computer to recognize different gestures you can make with your hands.

You will assemble a collection of different machine learning models so that they can work together to identify the shape of your hand in the webcam.



This project worksheet is licensed under a Creative Commons Attribution Non-Commercial Share-Alike License
<http://creativecommons.org/licenses/by-nc-sa/4.0/>

In this project, you will use pre-trained machine learning models that have been trained to find your hand in a view from your webcam.

These pre-trained models will give you the x,y coordinates of your wrist, and the top and bottom of each of your fingers.



For example:

	x	y
top of thumb	-82	2
base of thumb	-31	-107
top of index finger	-45	119
base of index finger	-25	-11
top of middle finger	-2	138
base of middle finger	9	-11

	x	y
top of ring finger	51	107
base of ring finger	39	-23
top of pinky finger	110	45
base of pinky finger	66	-46
wrist	17	-121

You will use these coordinates to see if your computer can learn to recognize different hand gestures, based on the coordinates of your fingers when you make those gestures.

For example:



fist



okay

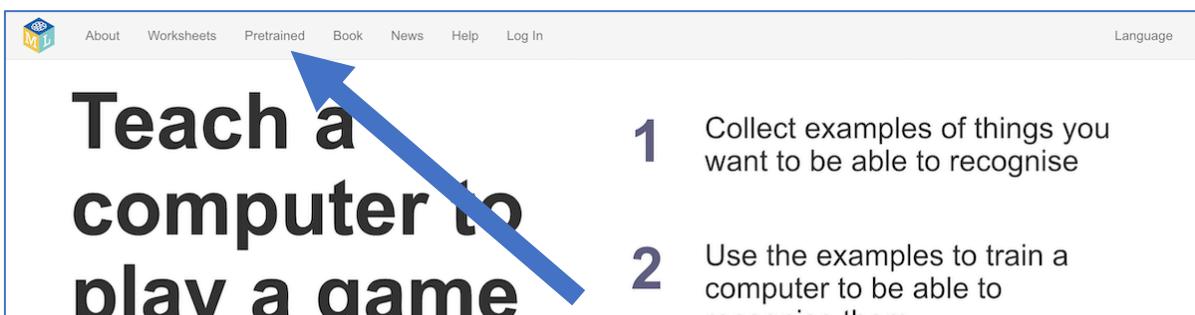


point

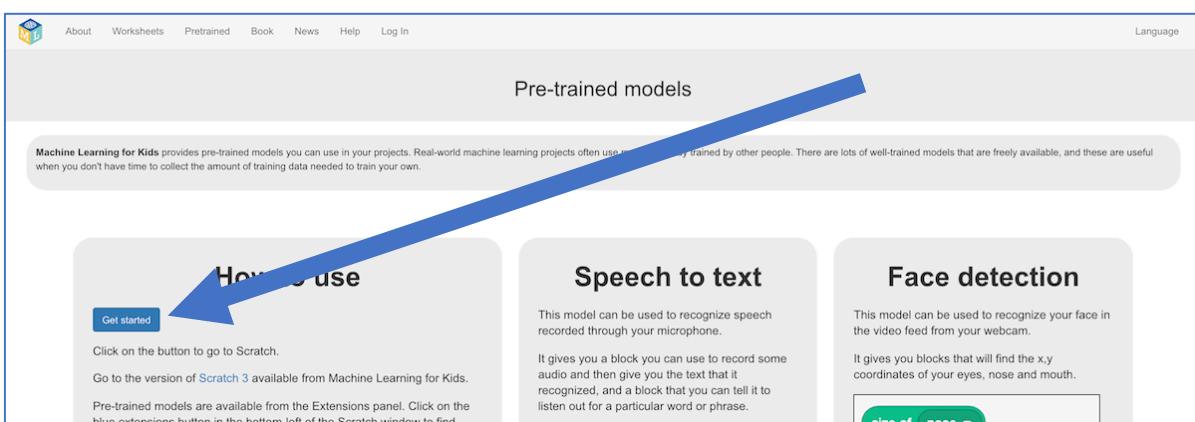
Let's start by trying out the pre-trained models to see what they can do.

1. Go to <https://machinelearningforkids.co.uk/> in a web browser

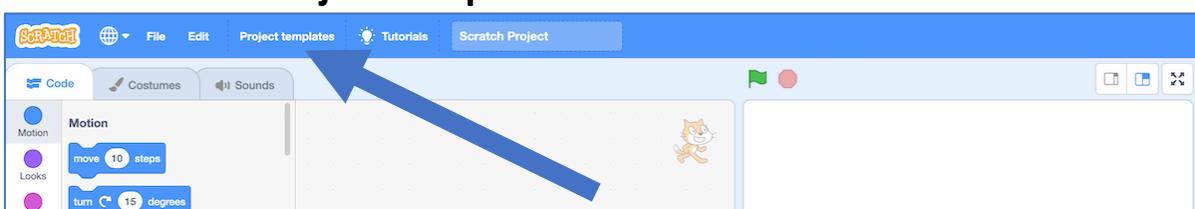
2. Click on “Pretrained”



3. Click on “Get started”



4. Click on “Project templates”



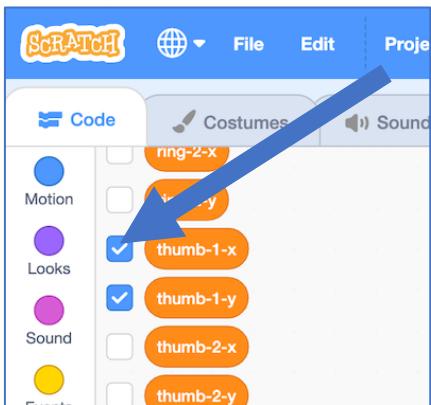
5. Click on the “Hand gestures” template

6. Click on the **Green Flag**

There might be a short delay while the machine learning model loads

7. Hold **one** hand up to your webcam

- 8.** Try ticking variables to see the values that the model gives you
Move your hand in the webcam to see how the numbers change

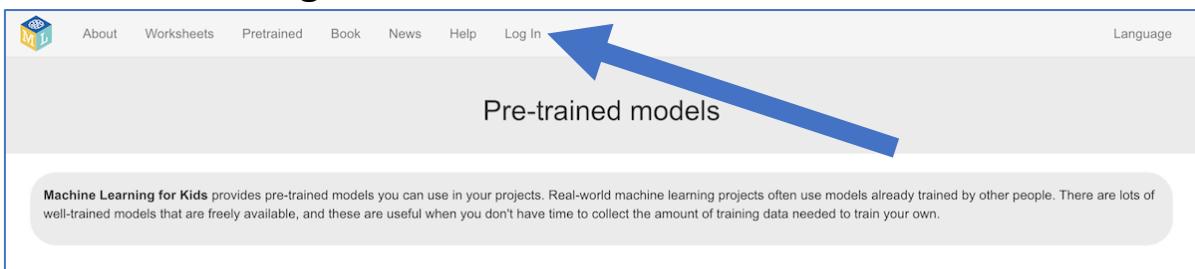


These are the numbers that you will get the computer to learn from to recognize different hand shapes and gestures.

Next, you'll prepare somewhere to collect these numbers as training examples.

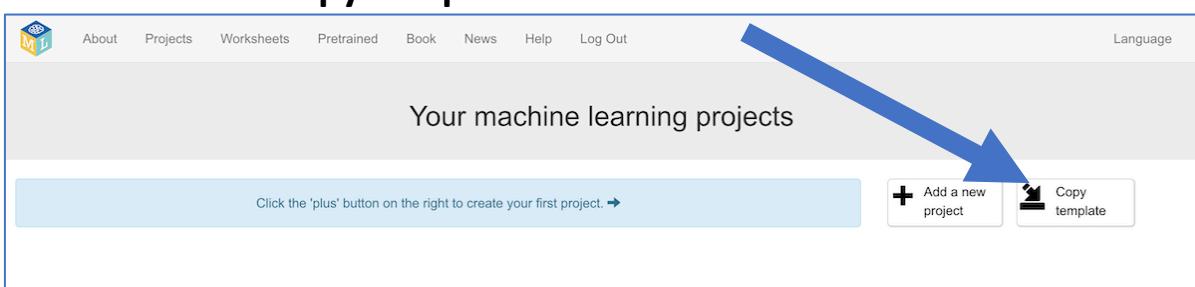
- 9.** Close the Scratch window
Go back to <https://machinelearningforkids.co.uk>

- 10.** Click on “Log In”

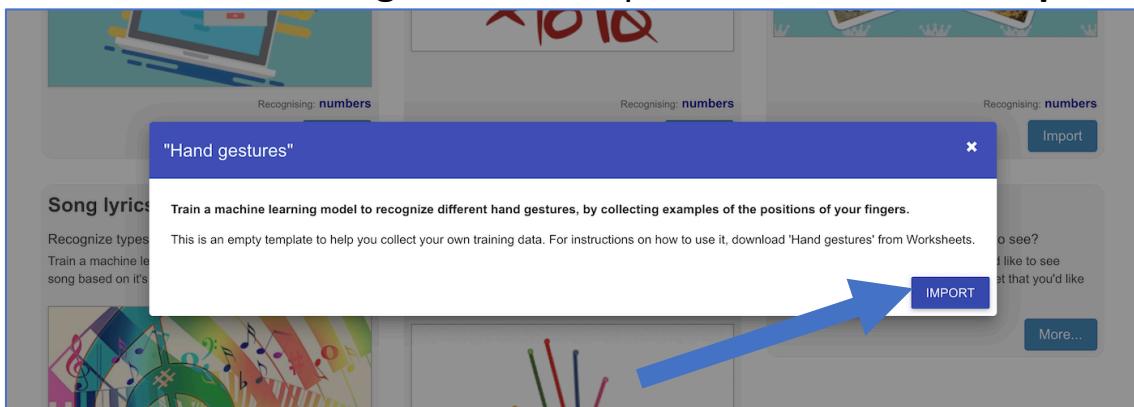


- 11.** Click on “Try it now”

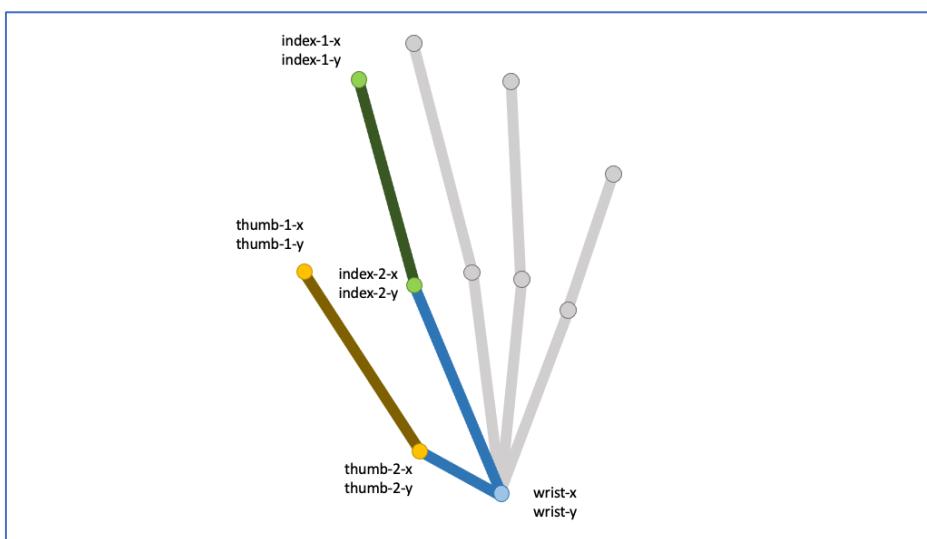
- 12.** Click the “Copy template” button



- 13.** Click the “Hand gestures” template, then click the “Import” button

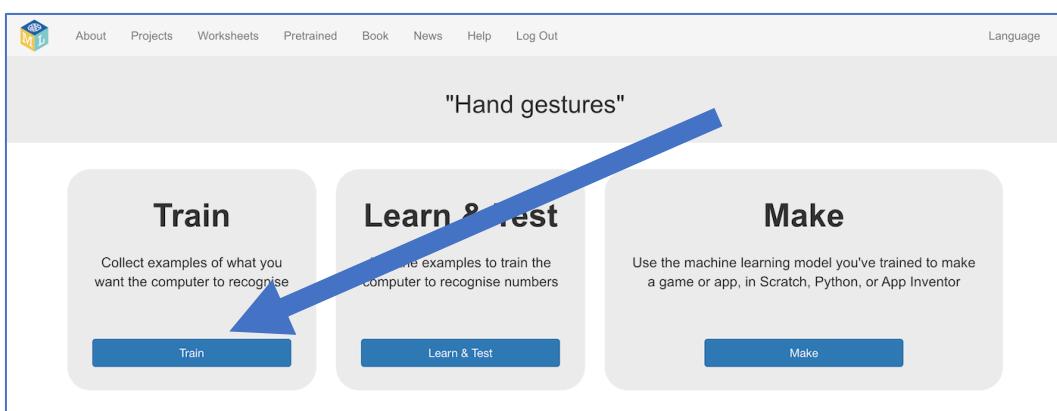


To make your Scratch coding simpler, we'll just be using the coordinates for two of your fingers. That will still give you ten numbers to use in your code!

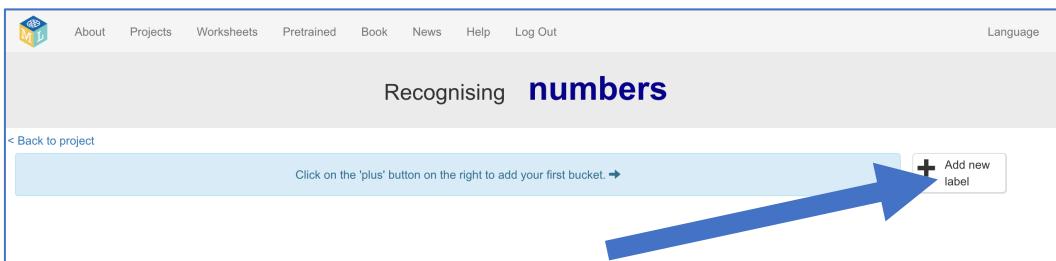


- 14.** You should see “Hand gestures” in your list of projects. Click on it.

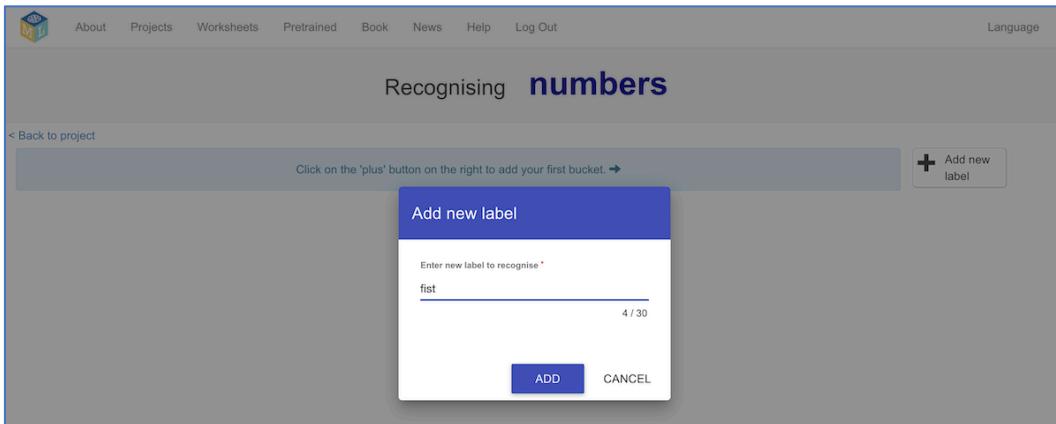
- 15.** Click the Train button.



16. Click on “+ Add new label”

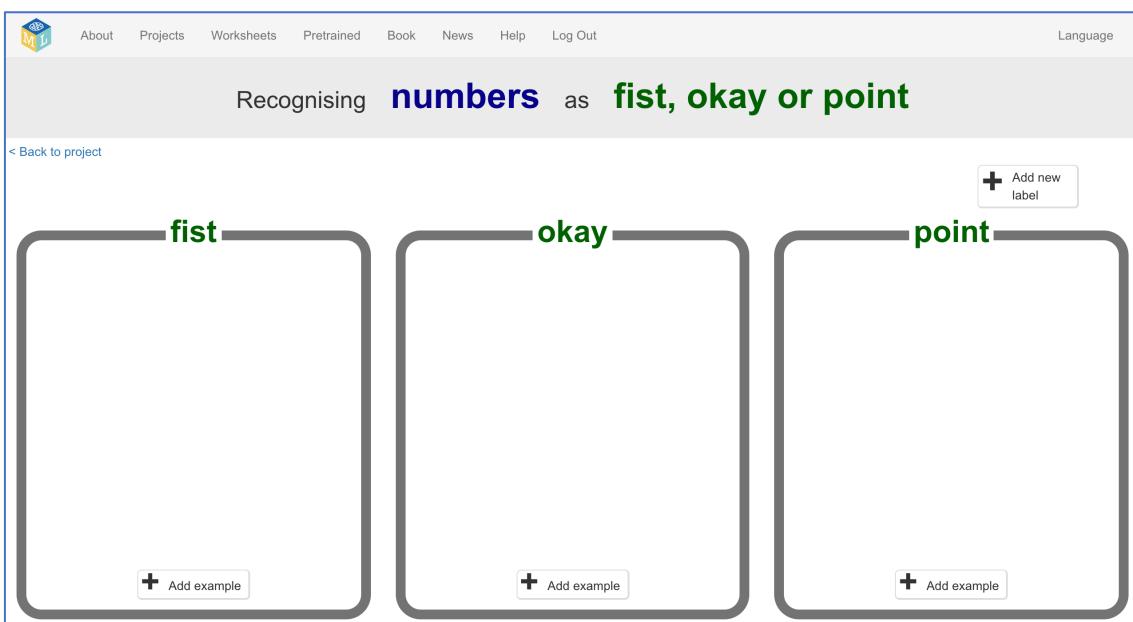


17. Type in “fist” and then click on “Add”.



18. Do that again to create a second bucket called “okay”. Do that a third time to create a bucket called “point”.

If you use different hand-gestures for your project, choose hand-shapes that can be recognised from the two fingers we’re using (thumb and index finger).



These are where you store training examples for the computer to learn from.

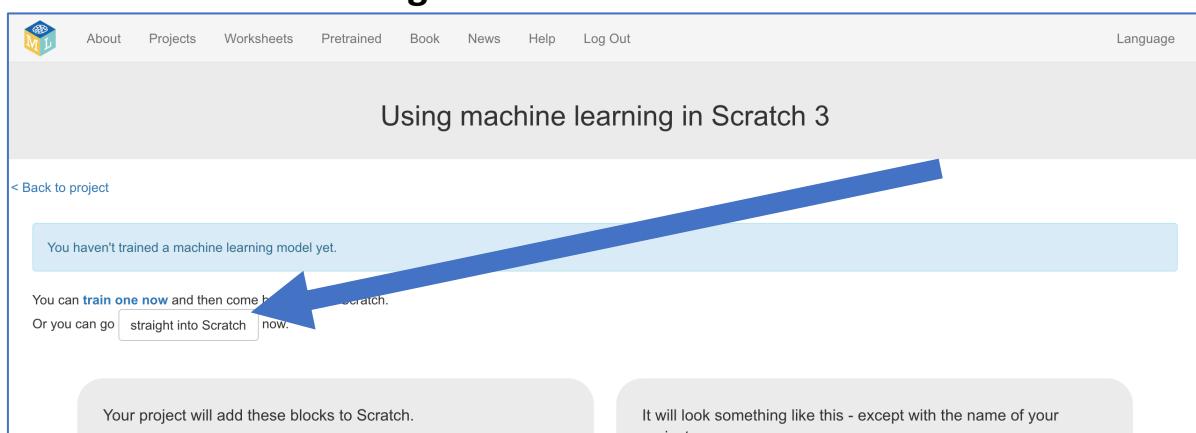
Next, you'll use Scratch to collect some examples.

19. Click on the “< Back to project” link

20. Click on the “Make” button

21. Click on the “Scratch 3” button

22. Click on the “straight into Scratch” button



23. Click on “Project templates” (as you did before)

The difference is that this time you will have extra blocks to use for your machine learning buckets.

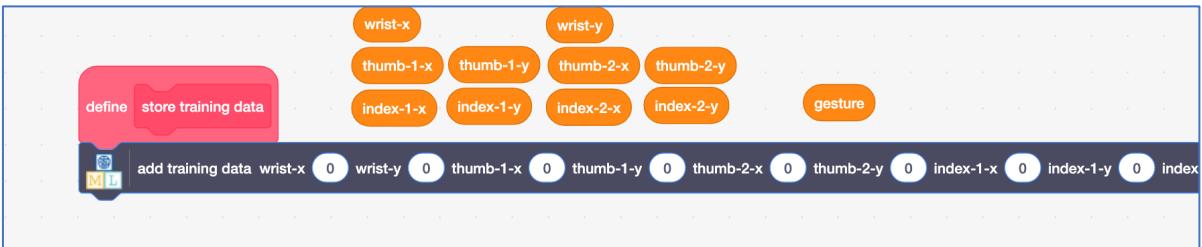
24. Click on the “Hand gestures” template (as you did before)

25. Find the placeholder for the “store training data” script

You should find it on the sprite called “main”



26. Add a block from your machine learning project to store examples



27. Copy in the orange variable blocks into the matching spaces on the "add training data" block

Do this slowly and carefully!

There are a lot of blocks to add. (This is why we're only using ten numbers for this project instead of all twenty-two)



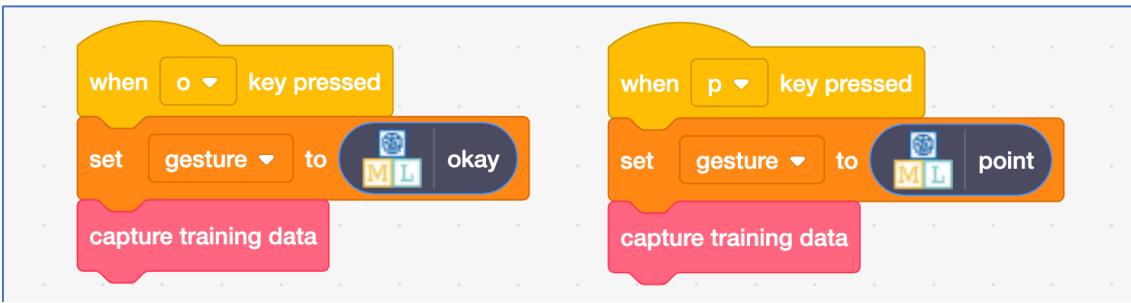
Don't forget to use the gesture block for the last menu space.



28. Create a script to collect a training example for your first gesture *I've used the "f" key because "fist" starts with f. If you're using different hand gestures for your project, you could pick a different key.*

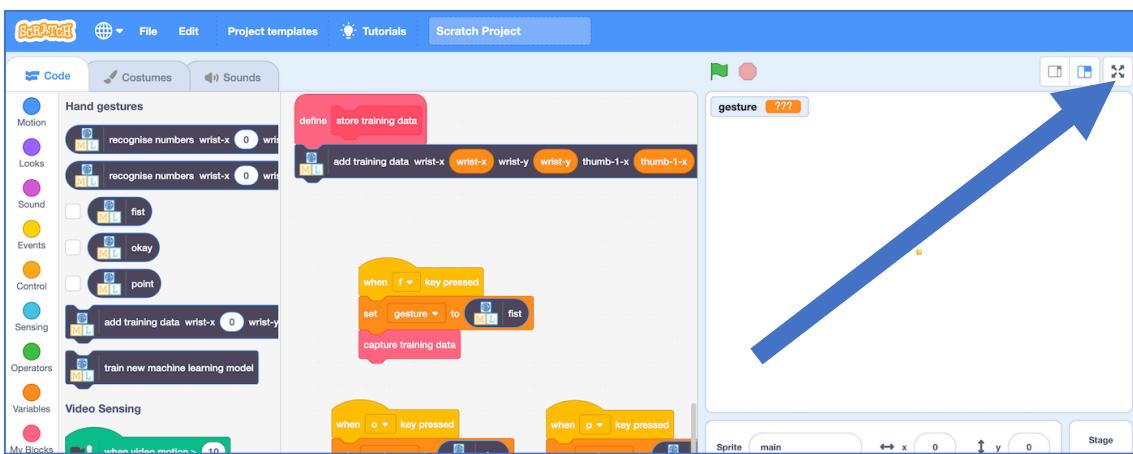


29. Create two new similar scripts for your other hand gestures



It's time to collect some training examples.

30. Click the full-screen button



31. Click on the Green Flag

32. Make your hand into the shape of a fist (or your first gesture) in the webcam, and wait until you think the dots have correctly found your hand *It works better if only one of your hands is visible in the webcam.*

33. Press the F button on your keyboard

(Or whichever key you picked for your first gesture.)

The dots stay still while coordinates are added to the “fist” training bucket

34. Repeat the last two steps (steps 32-33) nine more times

Try to vary the position of your hand in the webcam (for example, get some examples with your hand on the left side, and some on the right side) and vary how close your hand is to the webcam in each example.

35. Check the training tool page where you created the buckets to see the examples you've collected

Click on the “< Back to project” link and then click on “Train” (Don’t close the Scratch window!)

The screenshot shows the 'Recognising numbers as fist, okay or point' interface. There are three main sections: 'fist', 'okay', and 'point'. The 'fist' section contains 10 examples with detailed coordinate data for each hand gesture. The 'okay' section contains 10 examples. The 'point' section is empty. Each example includes a 'Add example' button.

36. Go back to the Scratch window

37. Collect training examples for your second hand gesture

As before, use your Scratch project but this time press the O key on your keyboard while making an “okay” gesture (see page 2 for an example). Collect ten training examples of this hand shape.

The screenshot shows the 'Recognising numbers as fist, okay or point' interface. The 'okay' section now contains 20 examples, each with a unique ID and a 'Add example' button. The 'fist' and 'point' sections remain empty.

38. Repeat for your final hand gesture

Use the **P** key to collect examples while pointing (see page 2 for example)
As before, try to get a variety of locations (where your hand is in the screen) and sizes (how close your hand is to the webcam) in your training examples.

The screenshot shows a web-based application for training a machine learning model. At the top, it says "Recognising numbers as fist, okay or point". Below this, there are three sections: "fist", "okay", and "point". Each section contains a grid of hand gesture examples with their corresponding coordinates. A "Language" dropdown menu is visible at the top right. A "Back to project" link is at the bottom left. A "Add new label" button is at the top right. Each section has a "Add example" button at the bottom right. The number "10" is centered at the bottom of each section.

Category	Example 1 (wrist-x)	Example 1 (wrist-y)	Example 2 (wrist-x)	Example 2 (wrist-y)	Example 3 (wrist-x)	Example 3 (wrist-y)	Example 4 (wrist-x)	Example 4 (wrist-y)	Example 5 (wrist-x)	Example 5 (wrist-y)	Example 6 (wrist-x)	Example 6 (wrist-y)	Example 7 (wrist-x)	Example 7 (wrist-y)	Example 8 (wrist-x)	Example 8 (wrist-y)	Example 9 (wrist-x)	Example 9 (wrist-y)	Example 10 (wrist-x)	Example 10 (wrist-y)
fist	100.91887196383067	-50.512761024468	124.17545423978603	-58.03085993713144	74.289455929776163	-39.5223716712655904	81.21413801425444	-54.1074989036778	81.21413801425444	-54.1074989036778	37.7715393089576	-37.7715393089576	81.21413801425444	-54.1074989036778	81.21413801425444	-54.1074989036778	81.21413801425444	-54.1074989036778	81.21413801425444	-54.1074989036778
okay	44.42260381383036	41.74769420774538	44.42260381383036	41.74769420774538	47.40538232377916	45.23750372282195	106.45303738513913	-54.1074989036778	106.45303738513913	-54.1074989036778	53.74485211298515	-39.24480116940859	53.74485211298515	-39.24480116940859	53.74485211298515	-39.24480116940859	53.74485211298515	-39.24480116940859	53.74485211298515	-39.24480116940859
point	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	52.4914571347390101	

You've collected enough training examples to get started.

Next, you'll get the computer to learn to recognize hand gestures by using these examples to train a machine learning model.

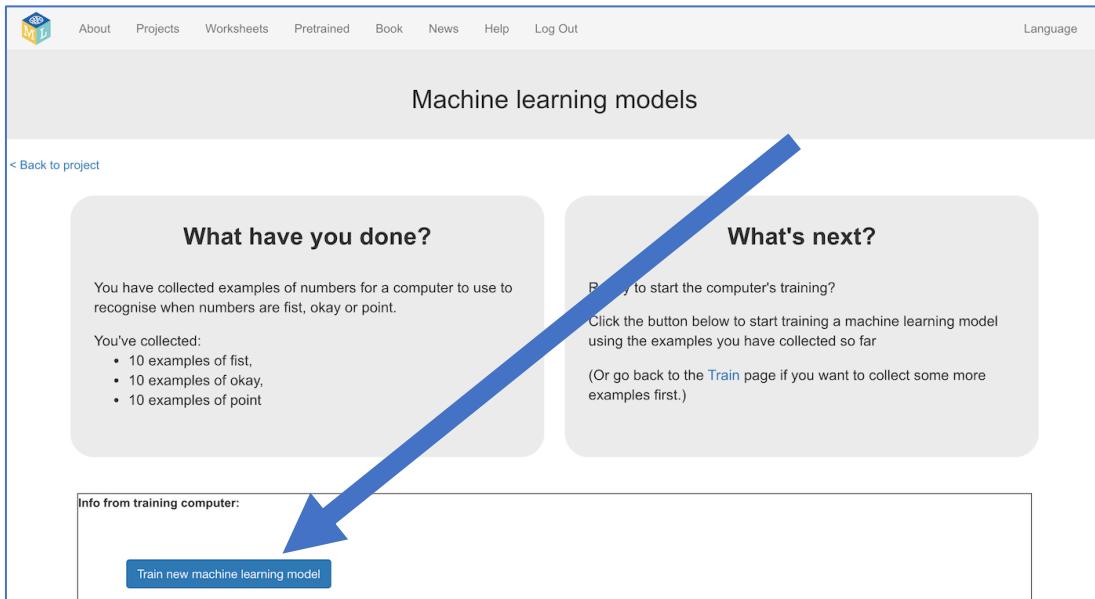
39. In the training tool window, click on the “< Back to project” link Leave the Scratch window open, as you'll be using it again in a moment.

40. Click on the “Learn & Test” button

The screenshot shows the "Learn & Test" button highlighted with a large blue arrow pointing towards it. The other buttons are labeled "Train" and "Make". The "Learn & Test" button is described as "Use the examples to train the computer to recognise numbers". The "Train" button is described as "Collect examples of what you want the computer to recognise". The "Make" button is described as "Use the machine learning model you've trained to make a game or app, in Scratch, Python, or App Inventor".

Train	Learn & Test	Make
Collect examples of what you want the computer to recognise	Use the examples to train the computer to recognise numbers	Use the machine learning model you've trained to make a game or app, in Scratch, Python, or App Inventor

41. Click on the “Train new machine learning” button

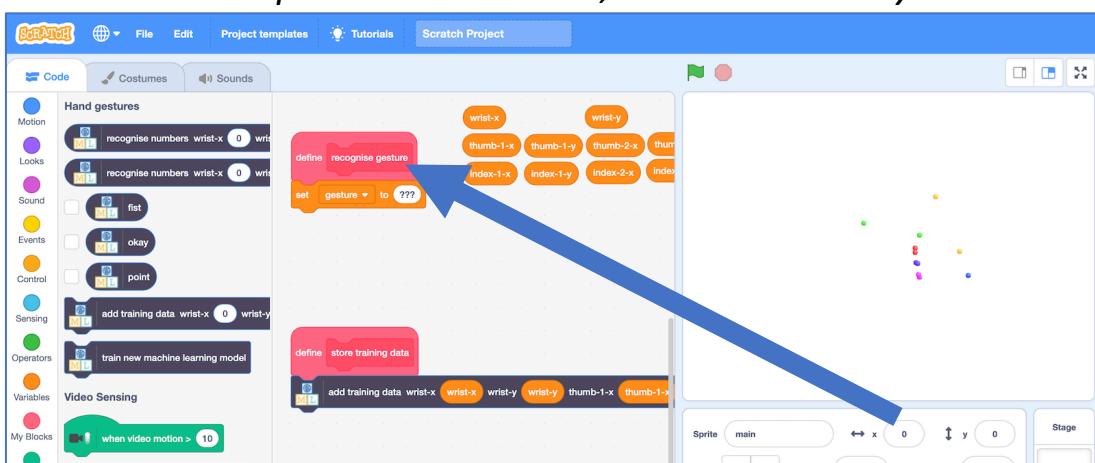


42. Go back to the Scratch window

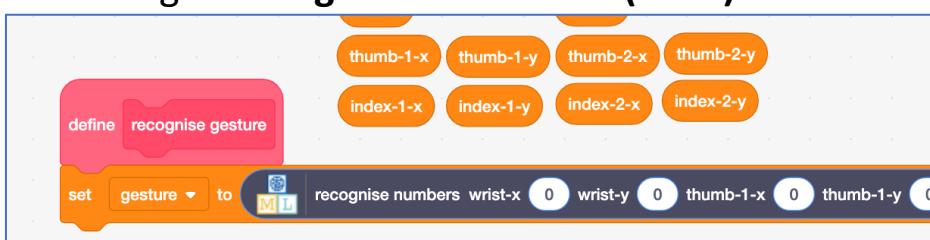
43. Click on the full-screen button so you can see the code again

44. Find the placeholder for the “recognise gesture” script

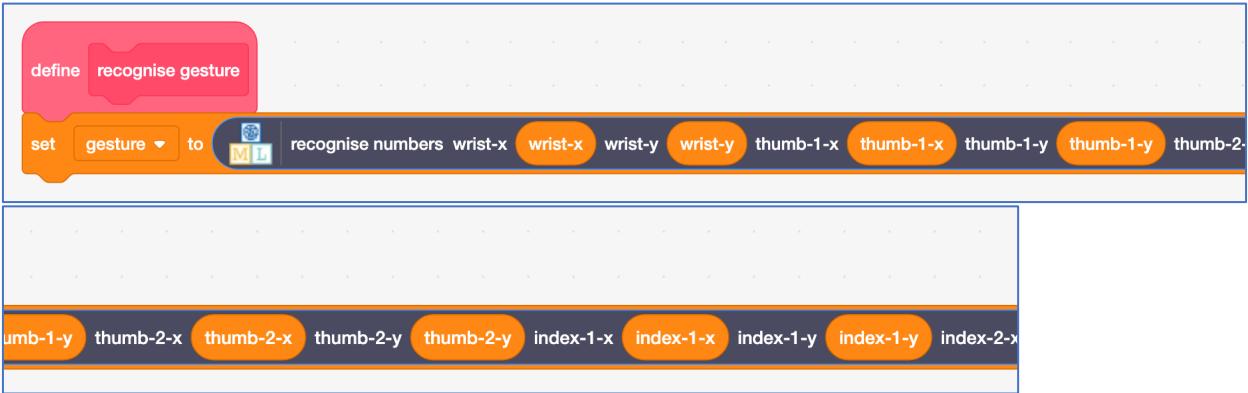
It is also in the sprite called “main”, above the code you worked on before.



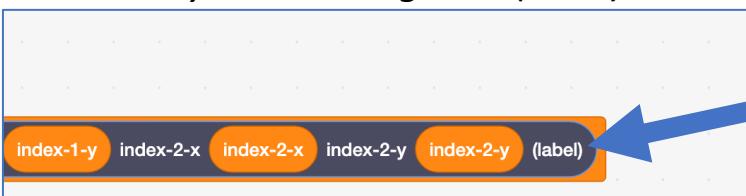
45. Drag a “recognise numbers ... (label)” block into the gap with ???



- 46.** Copy in the orange variable blocks into the matching spaces on the “recognise numbers” block
Do this slowly and carefully!



*Make sure you are using the “(label)” block, **not** the “(confidence)” block.*



Your project is now ready to go. It is time to test your machine learning model!

- 47.** Click the **full-screen** button
- 48.** Click on the **Green Flag**
- 49.** Make one of your gestures and hold your hand in front of the webcam, and wait until you think the dots have correctly found your hand
- 50.** Press the **space bar** on your keyboard
The machine learning model’s prediction will be displayed in the top-left.

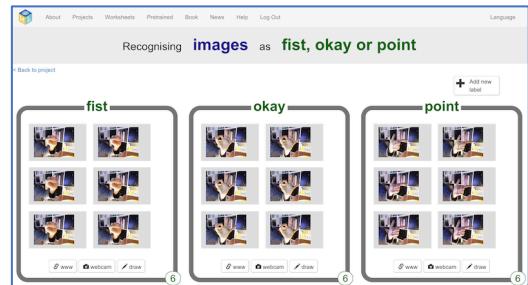
Does your machine learning model correctly recognize your hand gestures?

You might find that, with only ten examples of each shape, the model is still making mistakes.
If so, **repeat steps 31-42** (to collect twenty examples of each gesture and then train a new, improved machine learning model).

What have you done so far?

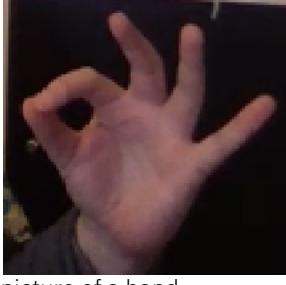
You've used a combination of different machine learning models (two pre-trained models, and one new custom model that you trained yourself) to make an artificial intelligence project that can do something new: recognize your hand gestures.

Instead of using a combination of machine learning models in this way, you could have just trained one new image machine learning model to recognize your hand shapes.



(If you'd like to see how that approach works, you can try the “Rock, Paper, Scissors” worksheet which uses this technique.)

Why do you think the technique you used in this project might be better?

Input	Model	Training challenge	Output
 picture of anything that includes a hand somewhere	Hand detection	Difficult Creating this model will involve a lot of training data with a large range of backgrounds (different types of indoors rooms, outdoor scenes and environments, etc.), with a variety of hands (different sizes, colours, etc.), in different positions Doing this is a lot of work!	 location of something that looks like a hand
 picture of a hand	Landmark detection	Difficult Creating this model will involve training data with a variety of hands in a variety of positions. Doing this is a lot of work!	 location of finger joints
 coordinates of finger joints	Gesture recognition	Simple Creating this model will involve recognizing relationship between a small collection of numbers. It doesn't need to deal with different backgrounds, or colours, or shadows, etc. This can be done with a small amount of training data.	“okay”



You used three machine learning models:

- A pre-trained hand detection model
- A pre-trained hand landmark detection model
- A custom gesture recognition model that you trained yourself

This is a common approach to machine learning projects:

- 1) break the overall thing that you want the project to do into separate steps
- 2) use pre-trained models for the steps that are common tasks
 - Tasks that other people have wanted to do (e.g. find a hand)
 - Tasks that other people have spent time and effort to train a machine learning model for
 - Tasks that aren't unique to your project
- 3) train a new custom model for the steps that are unique to your project

Doing all of this in a single model means that your model has to learn how to do all of these separate steps by itself, which means more training is needed.

Input	Model	Training challenge	Output
 picture of anything that includes a hand somewhere	Hand gesture detection	Very difficult This model needs to recognize a variety of hands (different sizes, colours, etc.) in a variety of locations, when seen against a large range of backgrounds (different types of indoor rooms, outdoor scenes and environments, etc.), and making a variety of shapes/gestures.	“okay”



What have you learned?

Pre-trained models are great. If you only use pre-trained models, this can limit you to making projects someone else has already thought of. AI projects combine pre-trained models with custom models to get the best of both worlds: the detailed training data of pre-trained models, and ability to make new models for new project ideas.

Ideas and Extensions

Now that you've finished, why not give one of these ideas a try?

Or come up with one of your own?

Try different hand gestures

Try making the project with your own ideas for hand gestures.

Try using an images model

Try making this project using a single, custom, images machine learning model. Which technique is easier for you to train? Which approach do you think works the best?

Use different finger coordinates

Instead of using locations for your wrist, thumb and index fingers, you could create the numbers project yourself and choose the number fields that you want to use.

Do you think you need the coordinates for the tip of each finger **and** the base of each finger? Would just the locations of fingertips be good enough?

Think about the hand shapes you want the computer to recognize and try making a project based on the values you think would best help the machine to know what hand shape you are making.