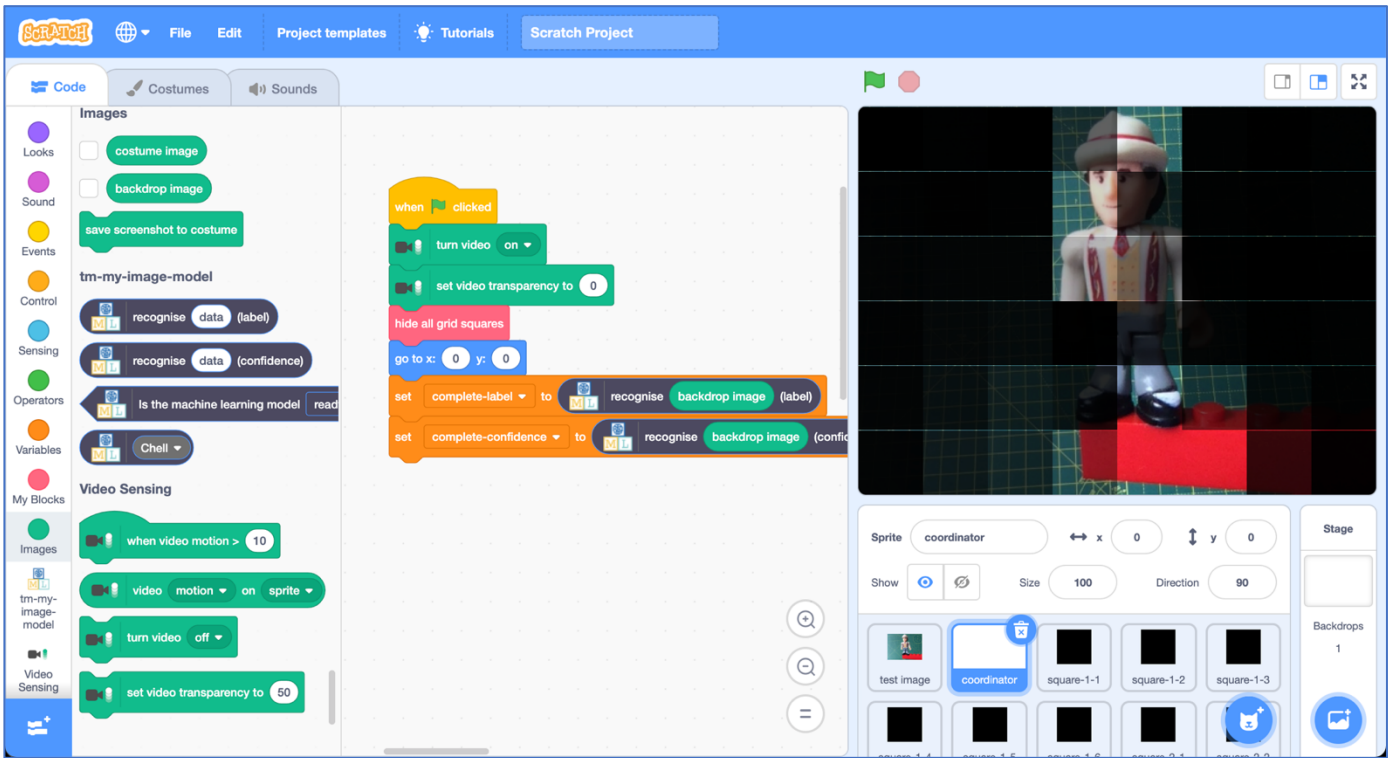
Explainable AI

You can train a machine learning model to recognise what is in an image. The model will only tell you what the overall picture is. It doesn't tell you the reason for the answer it gives you, or which parts of the image led it to give that answer.

In this project, you will learn a simple technique for understanding why an image classifier gives the answers that it does. You'll make a tool in Scratch that will help explain the parts of an image that your machine learning model recognized.



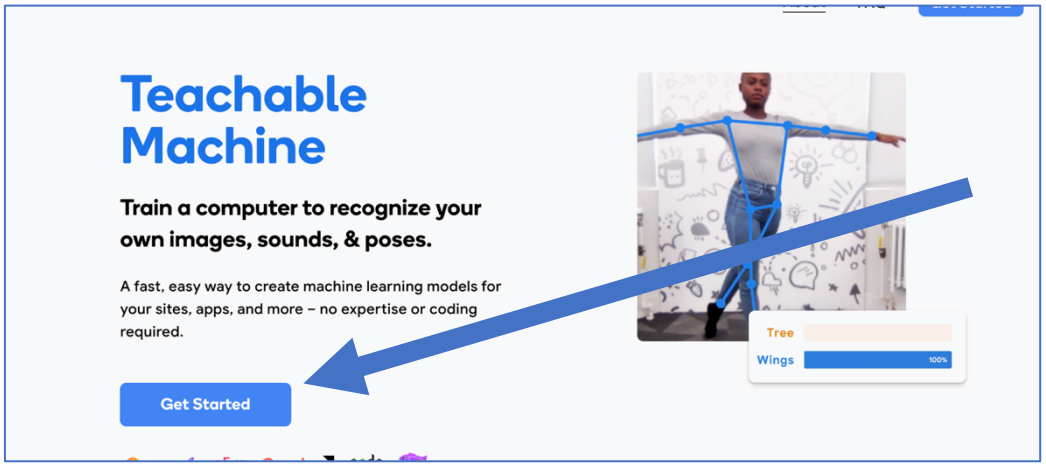
1. Choose four small objects to train the computer to recognise.

It helps if the objects are a similar size and have something in similar about their appearance.

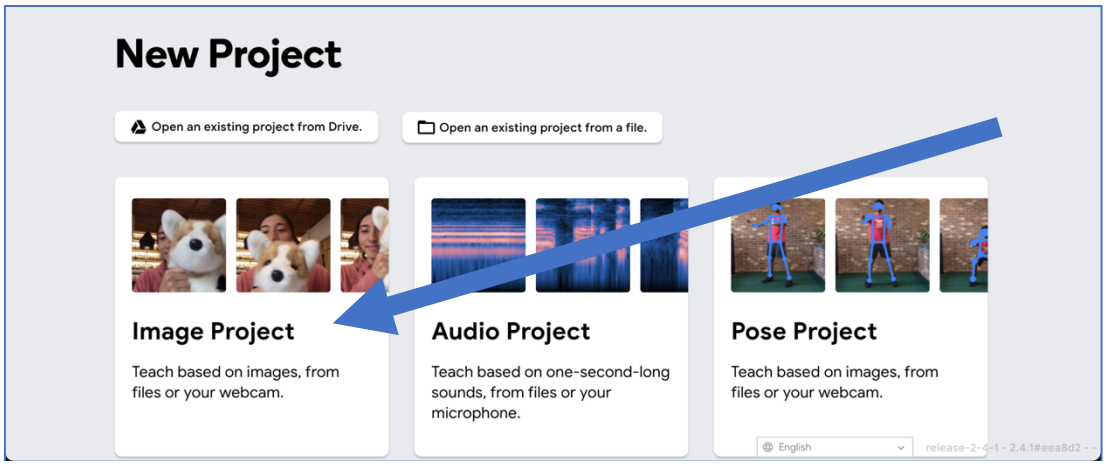
For example, I used four different character toy figures.

2. Go to https://teachablemachine.withgoogle.com in a web browser

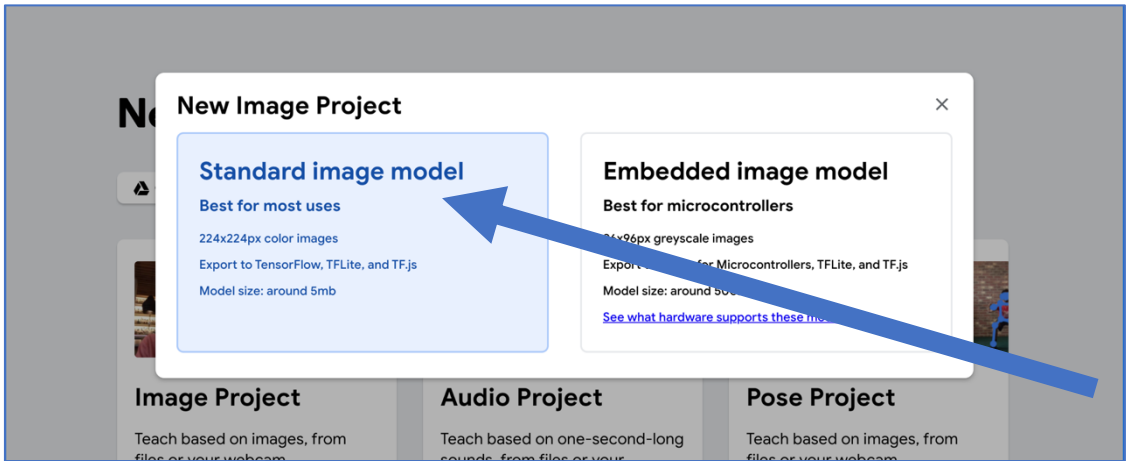
3. Click on "**Get started**"



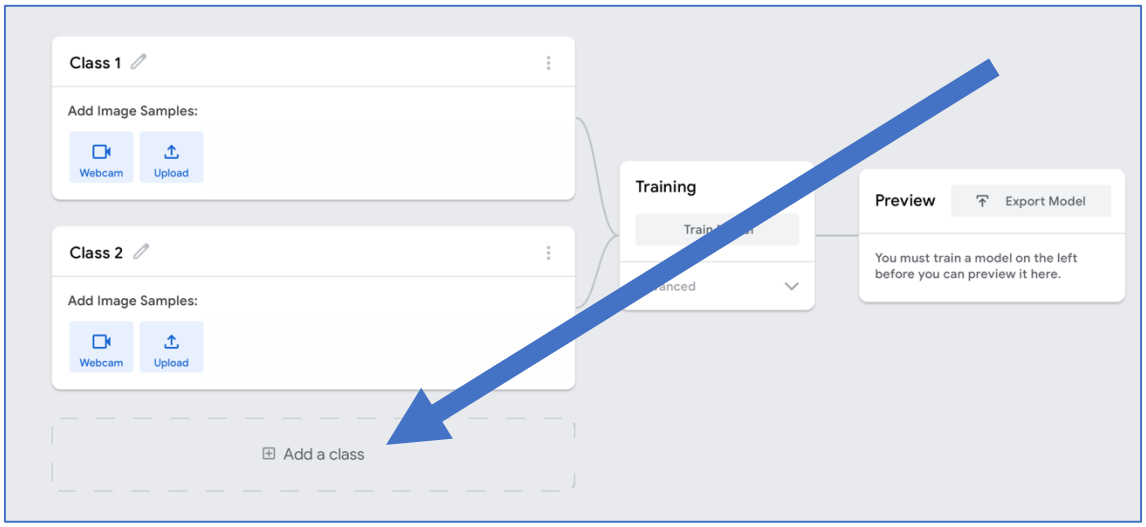
4. Click on "**Image Project**"



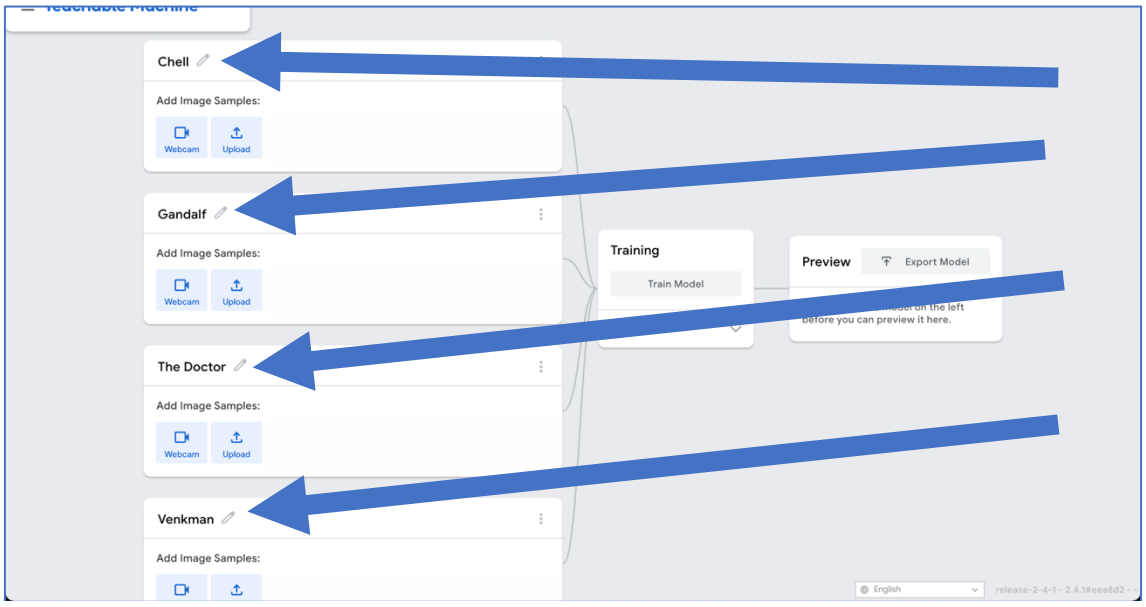
5. Click on “**Standard image model**”



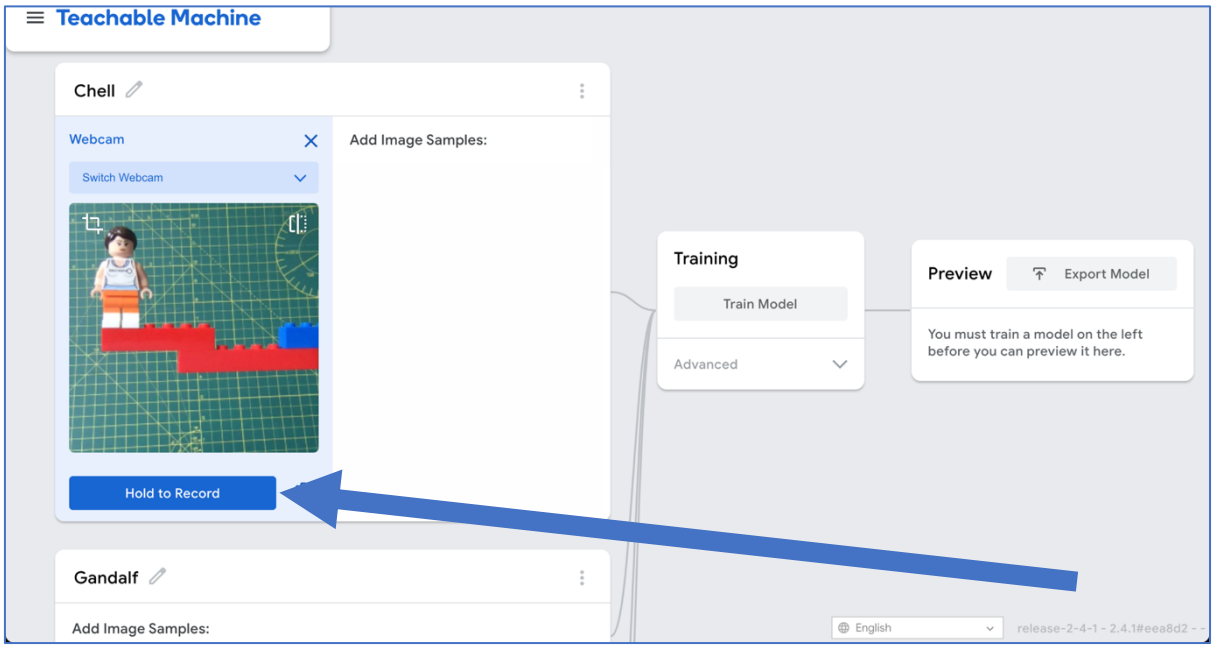
6. Click on the “**Add a class**” button twice, to create **four** classes



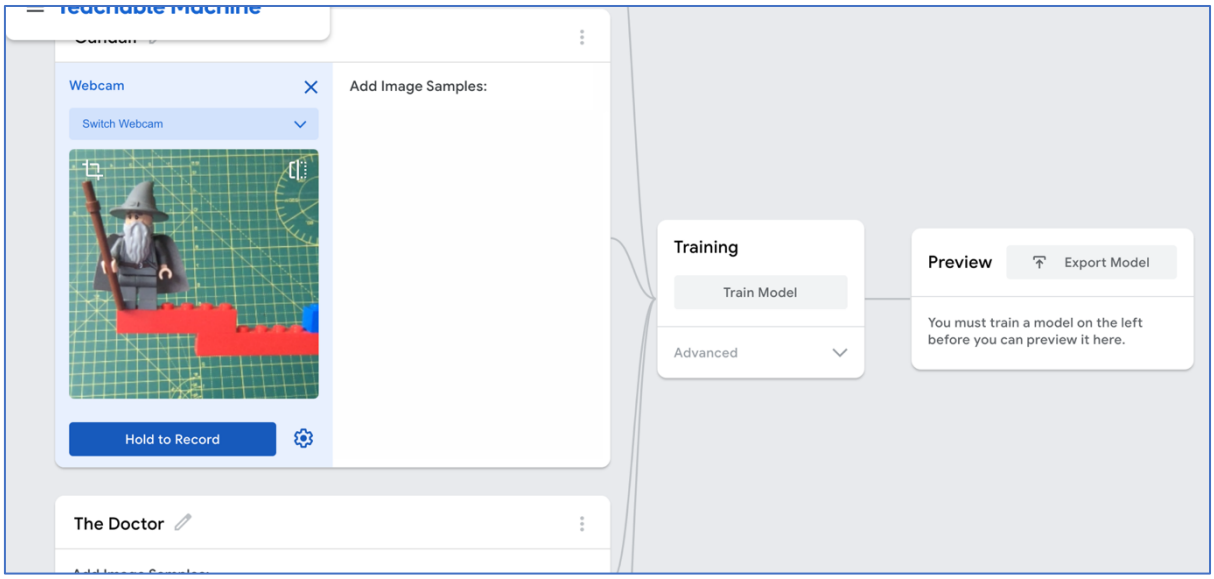
7. Click the **pencil** buttons to name the classes after your four objects



8. Use the "**webcam**" button to take photos of your object



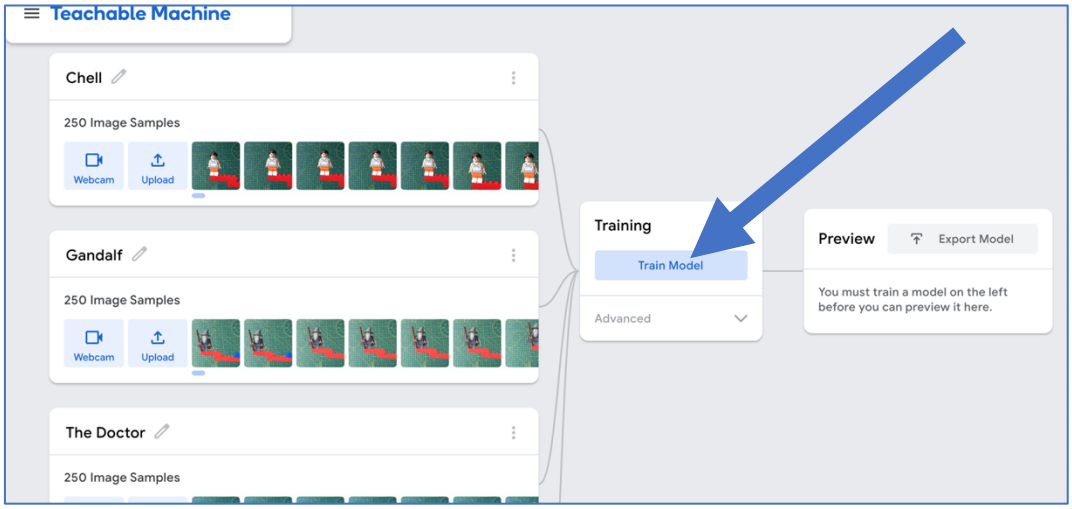
Make sure the background is identical in all of your photos.



Notice how I avoided having my fingers in the photos. If you can't do that, try to keep the position and shape of your hand consistent in all of your photos, so only the object is different.

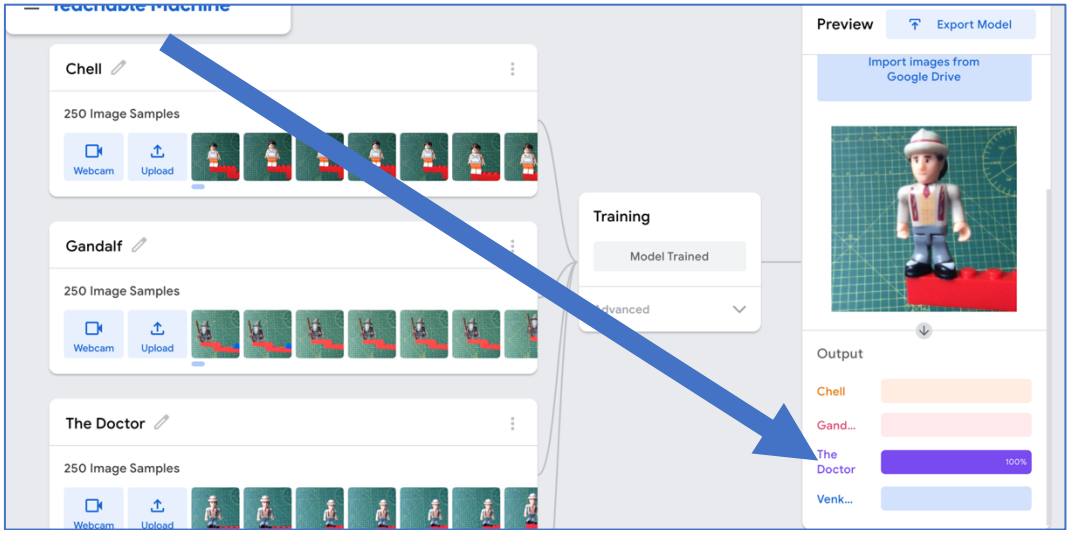
Important: You should vary the location of your object in all your photos. Move the object around in the webcam view while you hold down the "**Hold to Record**" button.

9. Once you have taken 250 images of each of your objects, click on the "**Train Model**" button

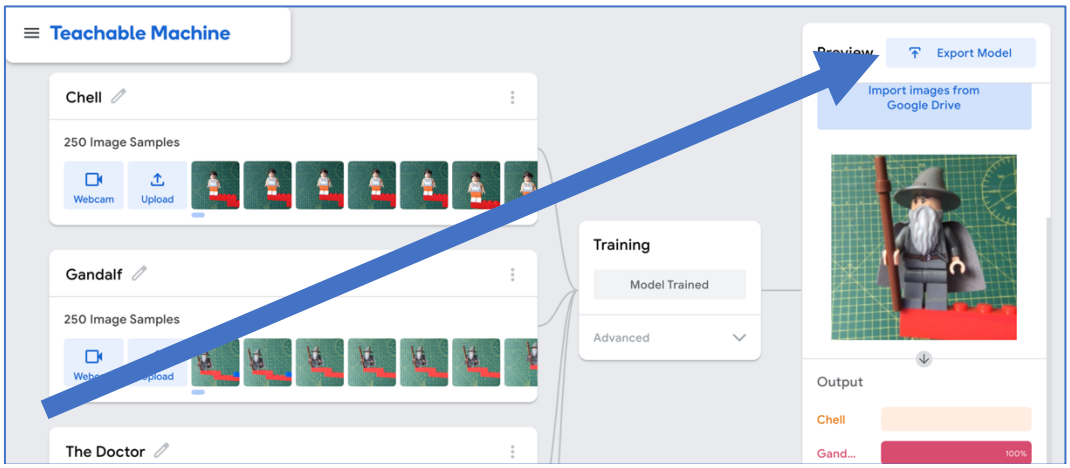


10. Use the "**Preview**" to check that your model is working well

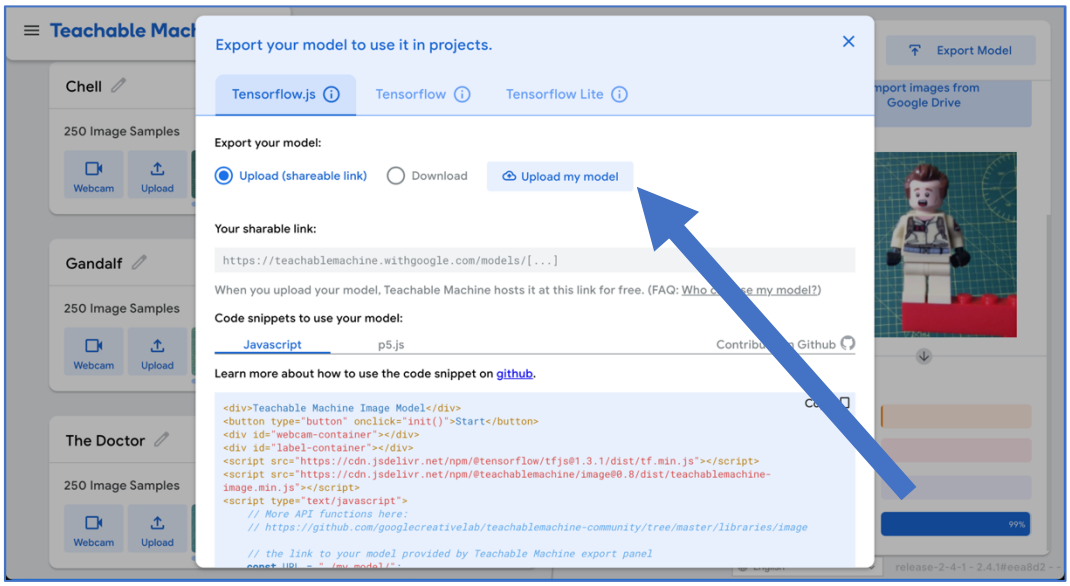
You can try adding more examples and training again if it isn't.



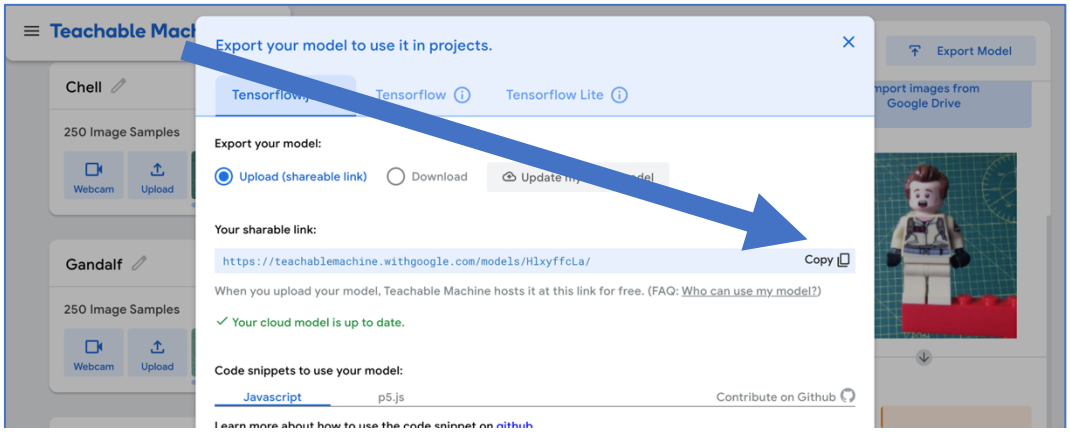
11. Click on "**Export Model**"



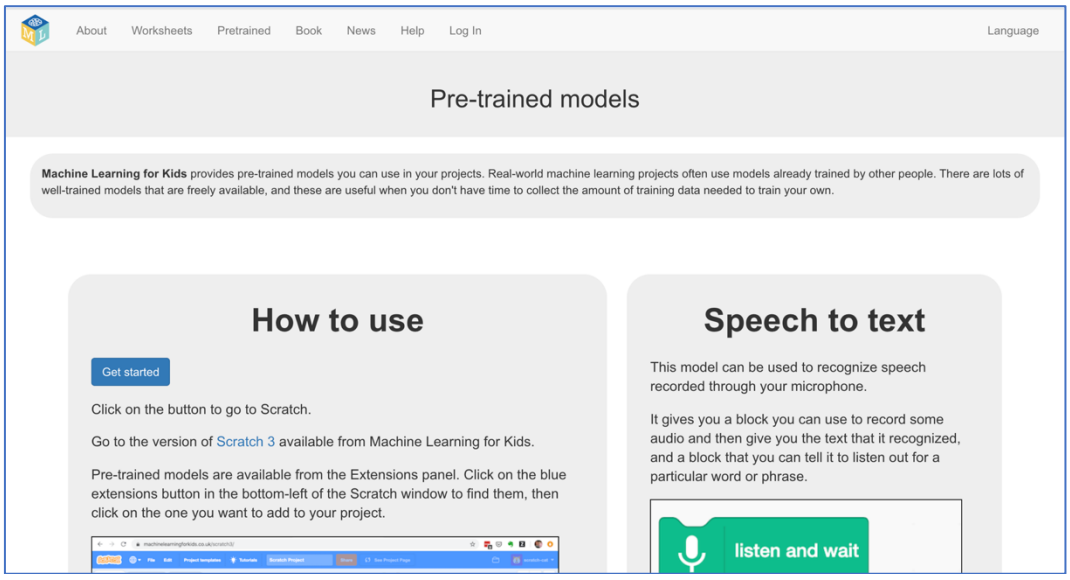
12. Click on “**Upload my model**”



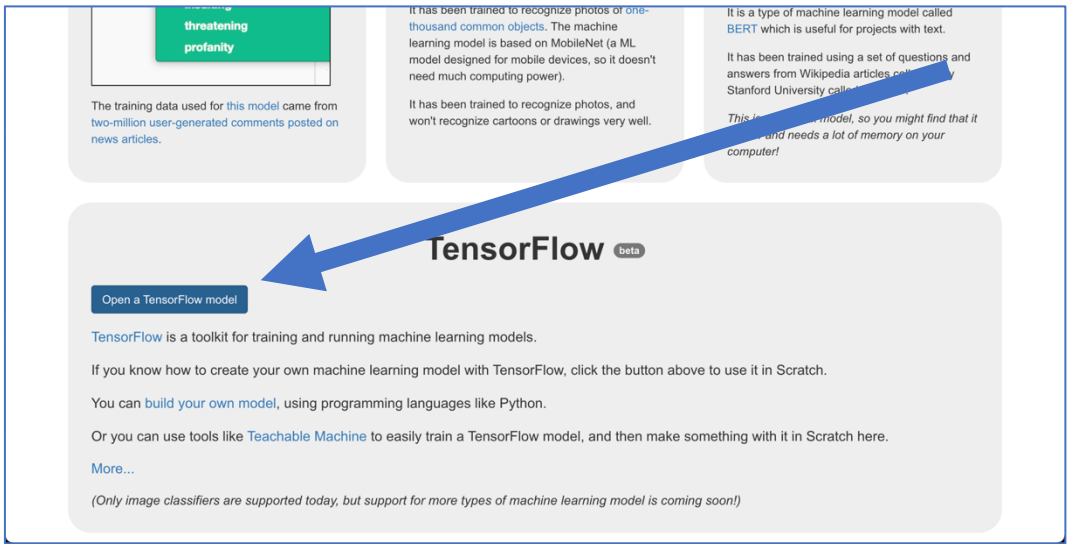
13. Copy the model link – you’ll need this in a moment



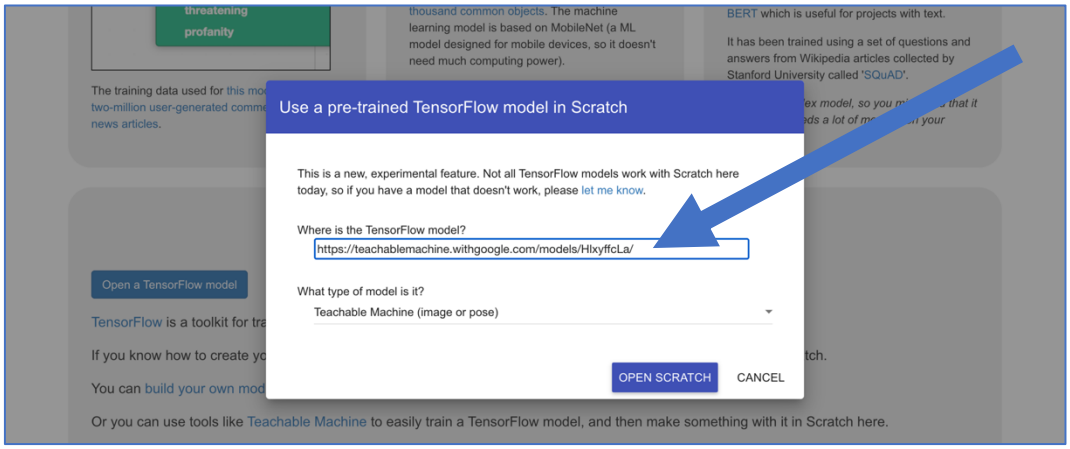
14. Go to https://machinelearningforkids.co.uk/pretrained



15. Click on the "**Open a TensorFlow model**" button

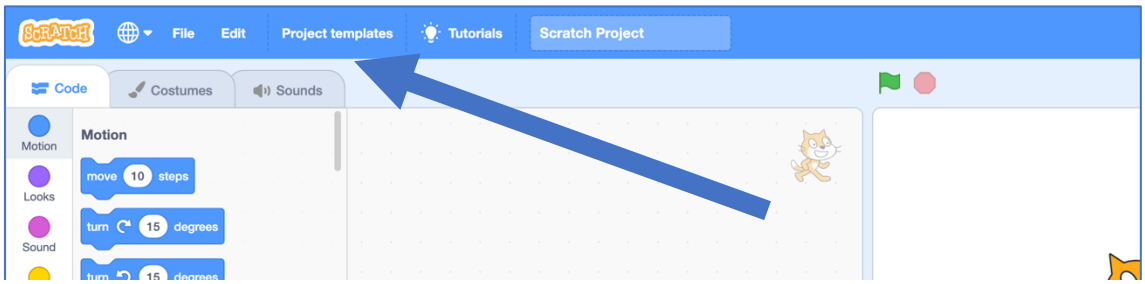


16. Paste in the link you copied in Step 13



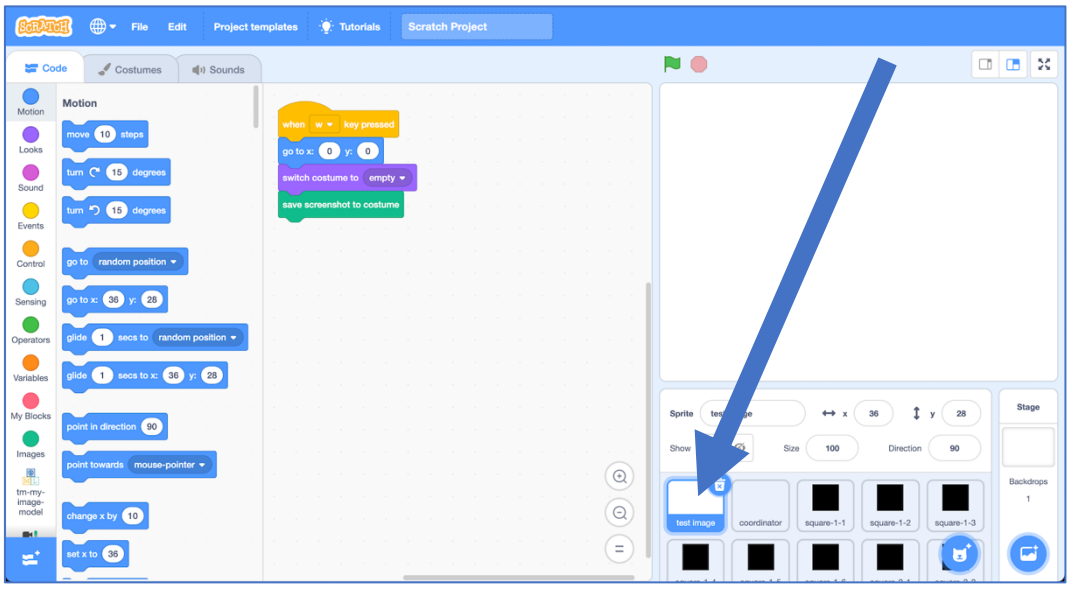
17. Click on "**Open Scratch**"

18. Click on the "**Project templates**" menu button



19. Click on the "**Explainability**" project template

20. Click on the “**test image**” sprite



21. Click on the Green Flag button

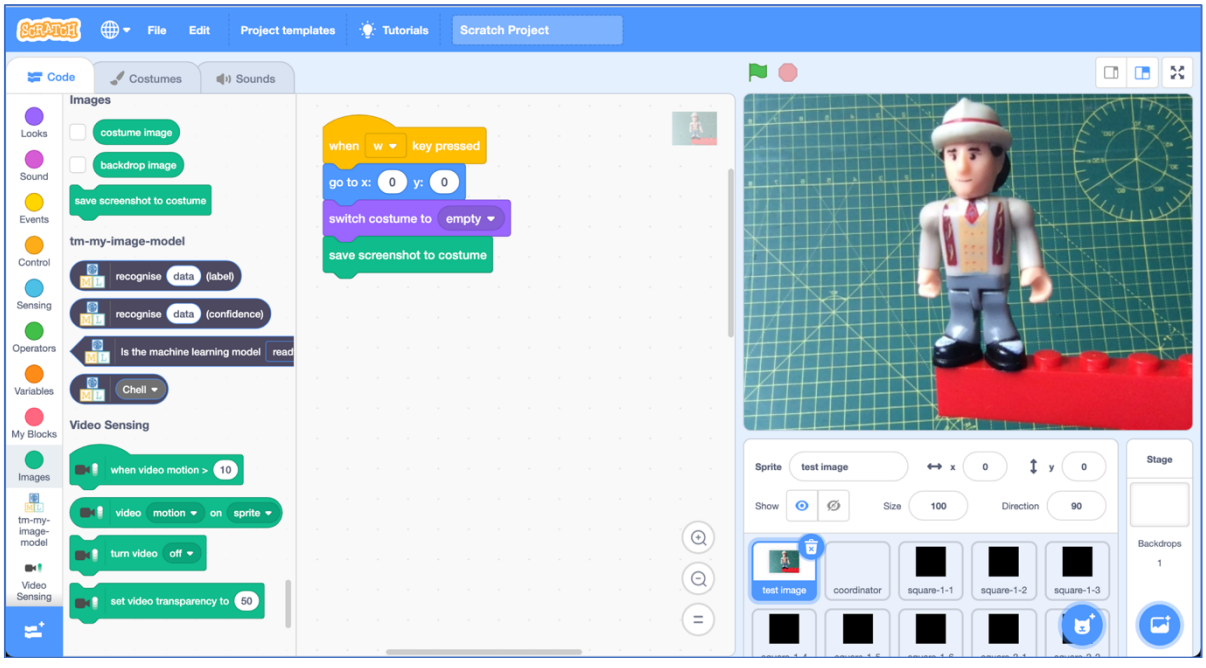
22. Hold one of your objects up to the webcam and press “w” on your keyboard to take a photo

Important: The background must be identical to the background you used for your training photos.

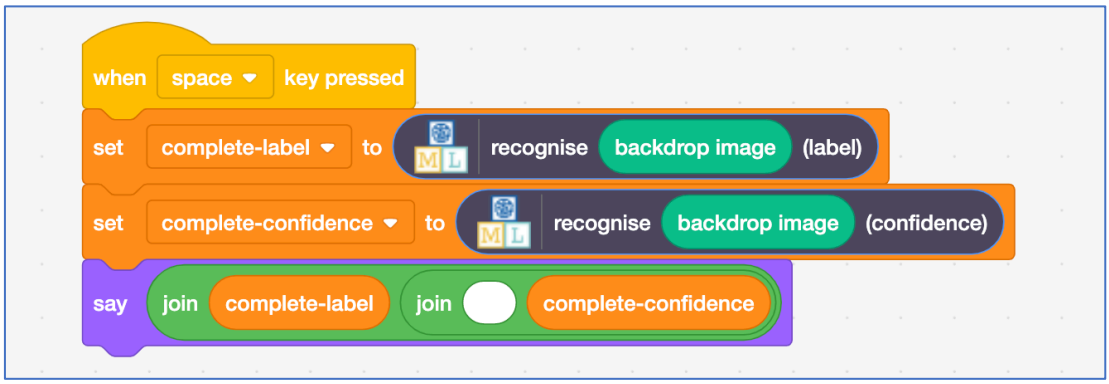
Try to position your object centrally.

Try to hold your object close enough to the webcam so that it fills a lot of the picture.

If you’re not happy with the photo, press “w” again to take a new one.

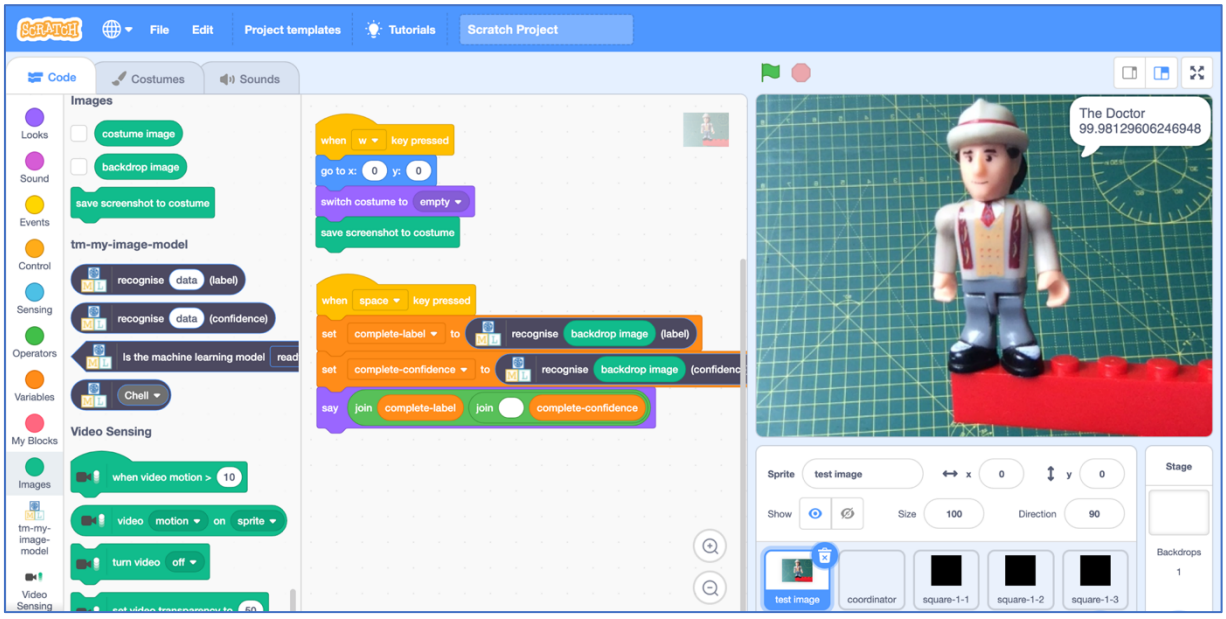


23. Create the following script



24. Press your spacebar

Make a note of the confidence score – you will need it again later.



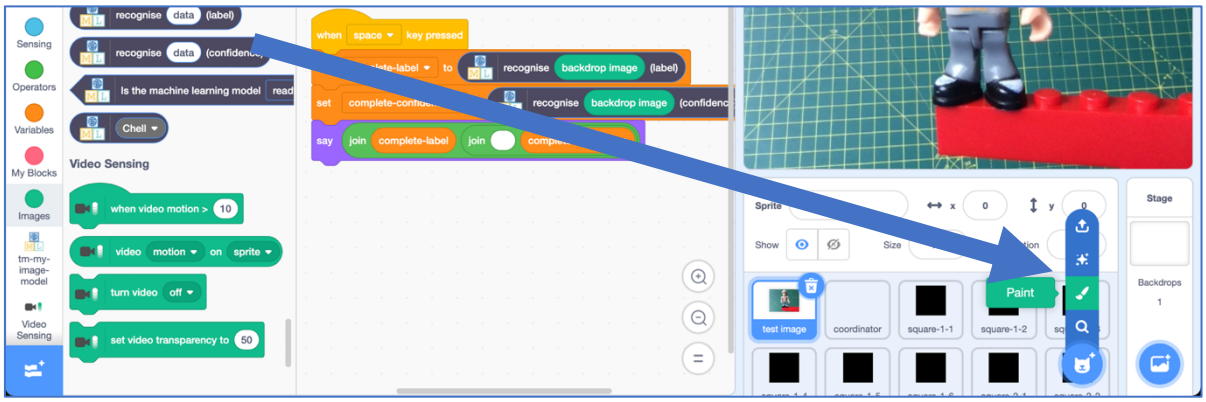
What have you done so far?

You've trained a machine learning model to recognize images of a few objects. The machine learning model can tell you it's prediction of what it thinks is in an image, but it doesn't tell you **why** it made that prediction.

It doesn't tell you what parts of the image were significant for the prediction, and what parts of the image the model thought were irrelevant.

Next, we'll see how you can learn a little about what your model thinks is most important in your test image.

25. Create a new sprite using the "**Paint**" option



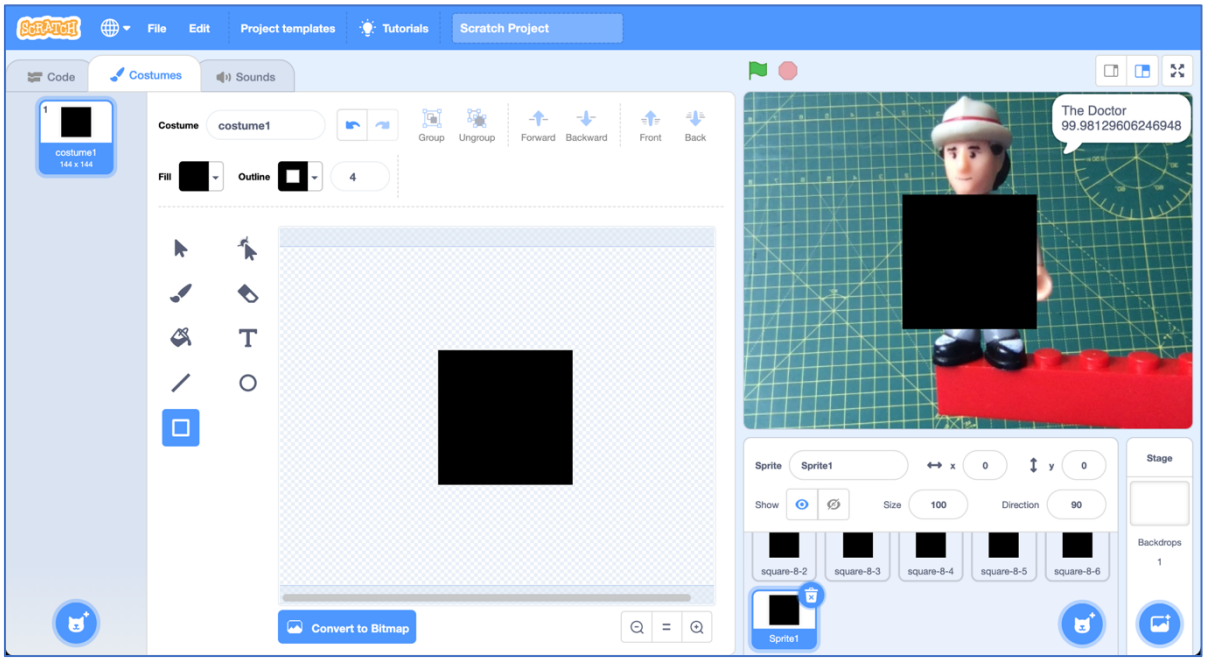
26. Draw a solid filled square

Choose a colour that won't give a hint to your machine learning model to choose one of the objects.

** a colour that isn't uniquely used by one of your objects.*

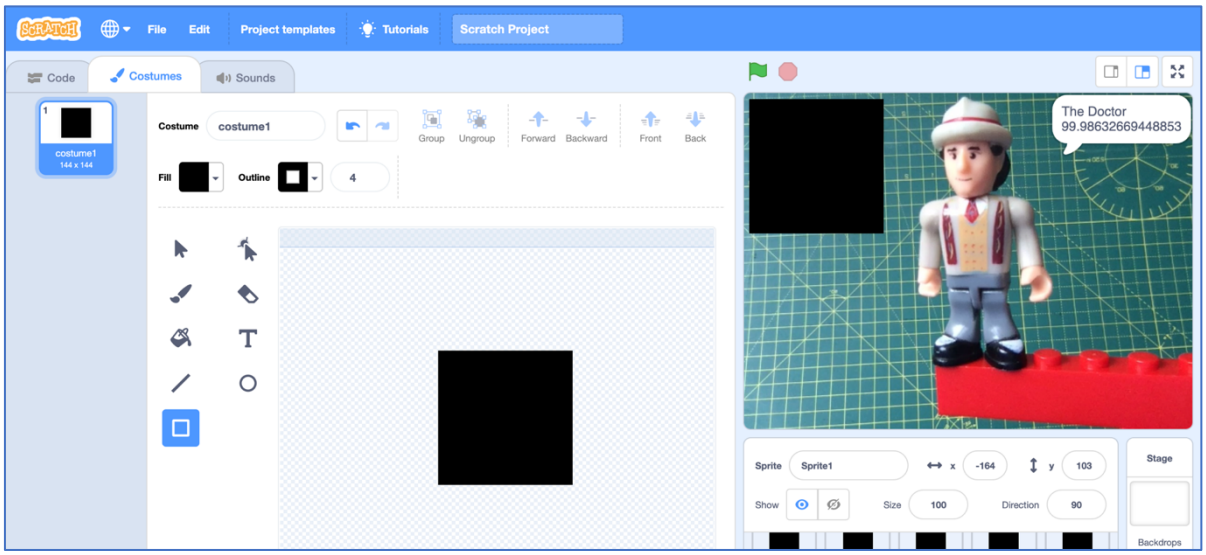
** a colour that is in none of your objects, or used equally in all of them.*

For example, I chose black because none of my four toy characters are mostly black.



27. Drag the sprite to a position on the stage that is as far away from your object as possible

28. Press the spacebar again

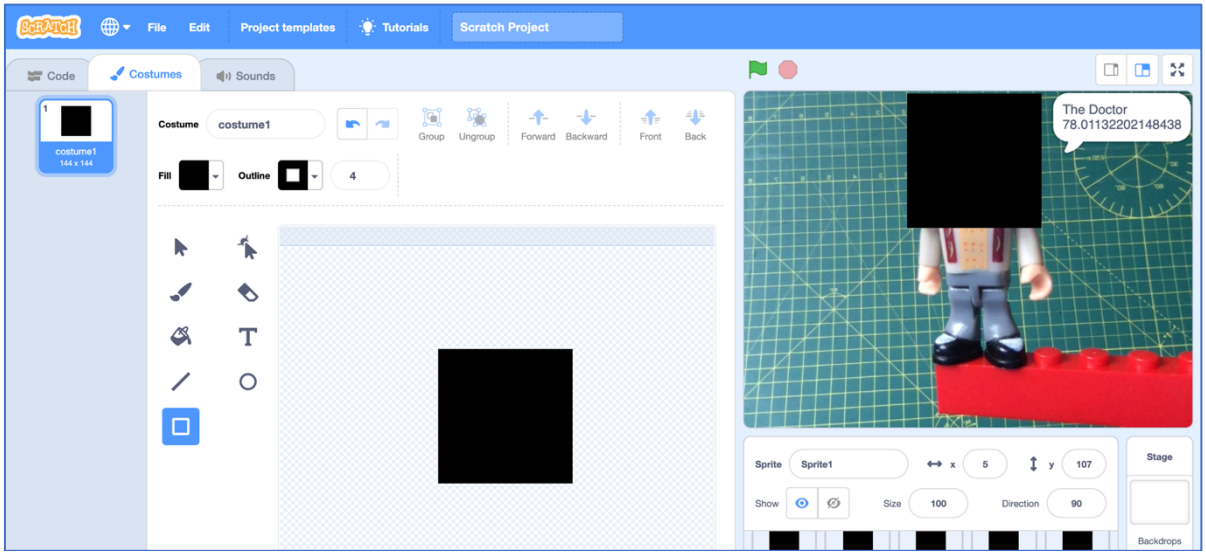


29. Compare the prediction your machine learning model makes with the prediction from Step 24

The confidence level should be very similar to the confidence from Step 24.
Covering up this area has not made much difference to the prediction.
The area you covered up was not very significant to the prediction.
The contents of that square did not have much to do with why the model thought this image looked like your object.

30. Move the square sprite to a position that covers something you think is going to be very significant

31. Press the spacebar again

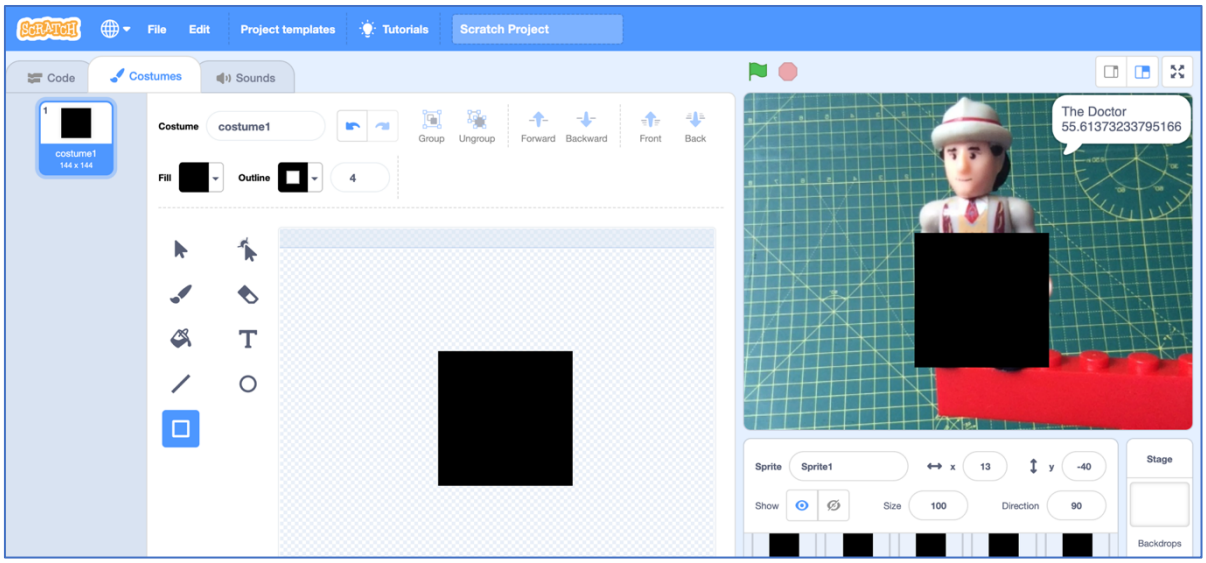


32. Compare the prediction your machine learning model makes with the prediction from Step 24

The confidence level should be different to the confidence from Step 24.
Covering up this area made a difference to the prediction.
The area you covered up was significant to the prediction.
The contents of that square had something to do with why the model thought this image looked like your object.
The model might still have recognized the object correctly, but without the area you covered up, it wasn't as confident in the prediction.

33. Repeat Steps 30-32

Try to find a position that makes the biggest difference on the machine learning model's prediction



If you find this difficult…

If every part of your object is visually unique and distinctive, then there will always be something for your machine learning model to recognize – this can mean that covering up one part of the image doesn't make a big difference to the confidence.

If this happens, you can:
** press "w" to take a new photo of a different one of your objects*
** make your square sprite larger so it covers even more of your image*

Because all of my objects were toy character figures, they all had something in common, so covering up distinctive features did make a difference to the confidence.

But if you trained your machine learning model really well, then it might be difficult to fool your model easily!

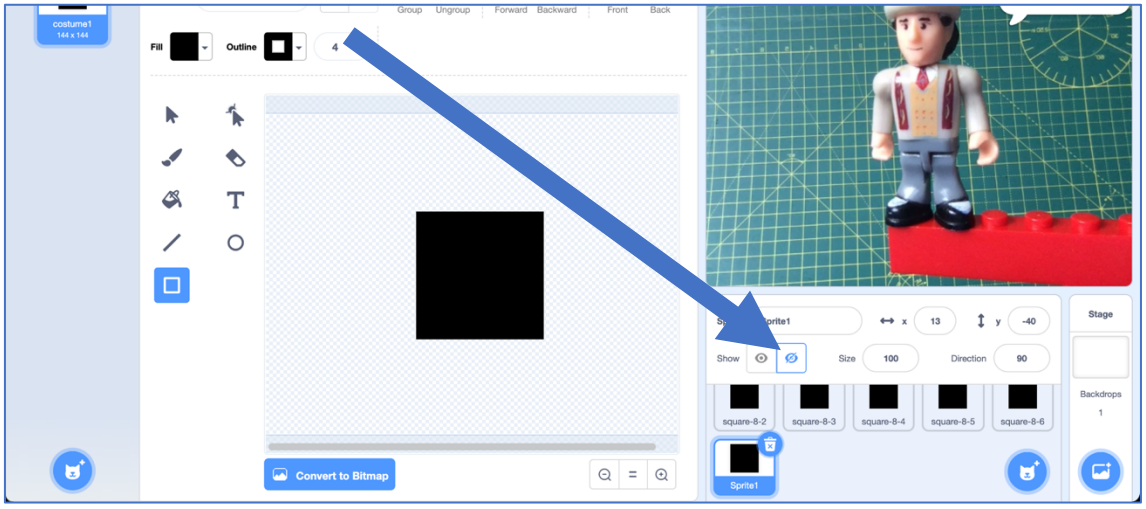
What have you done so far?

You've seen that although a machine learning model makes a prediction for an image as a whole, different areas of the image have different levels of significance to the prediction.

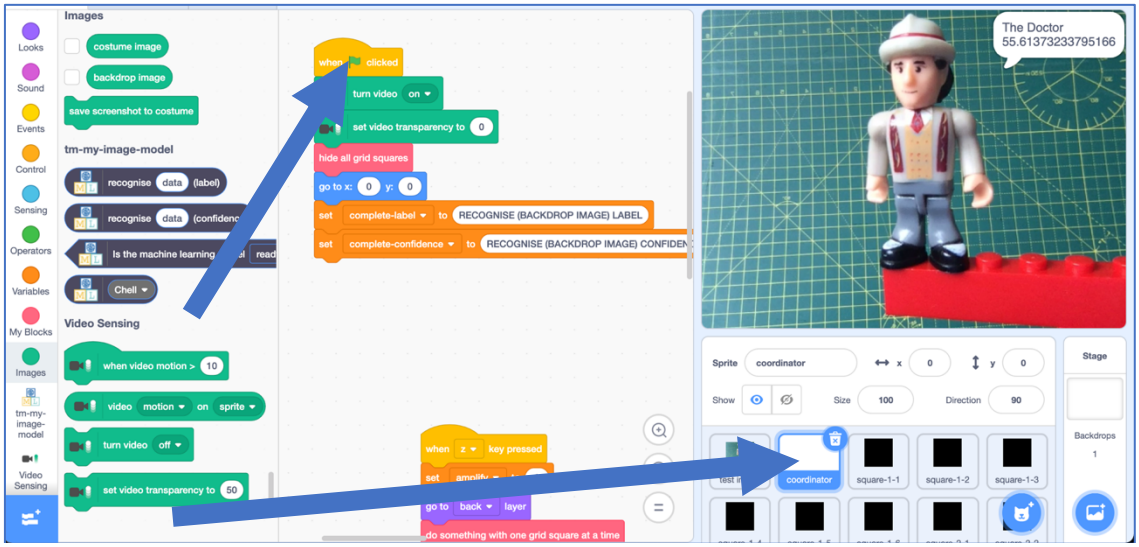
You've seen a simple way to measure this is to cover parts of the image and see the difference that it makes to the confidence the model has.

Finally, you will try a more organised way to use this technique – moving the cover square to every possible location and seeing the difference it makes in each position.

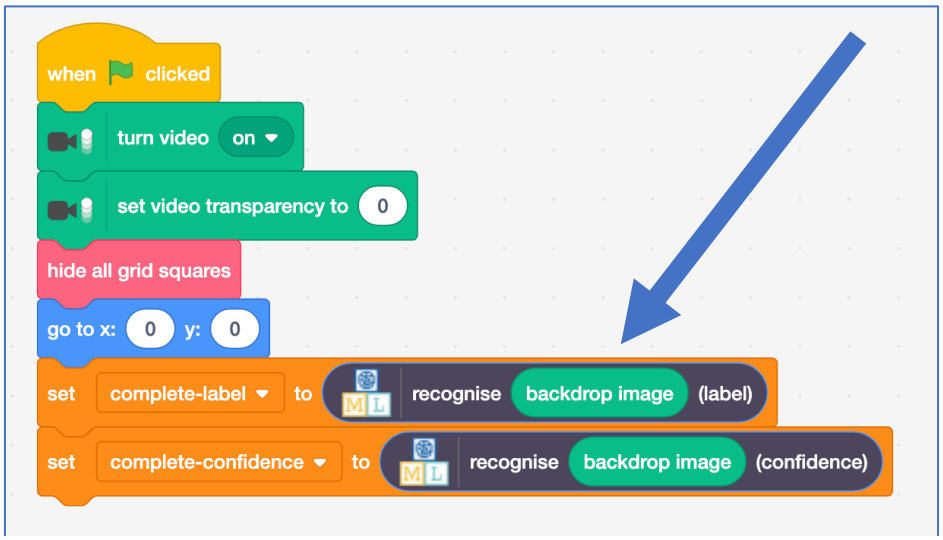
34. Hide your black square sprite



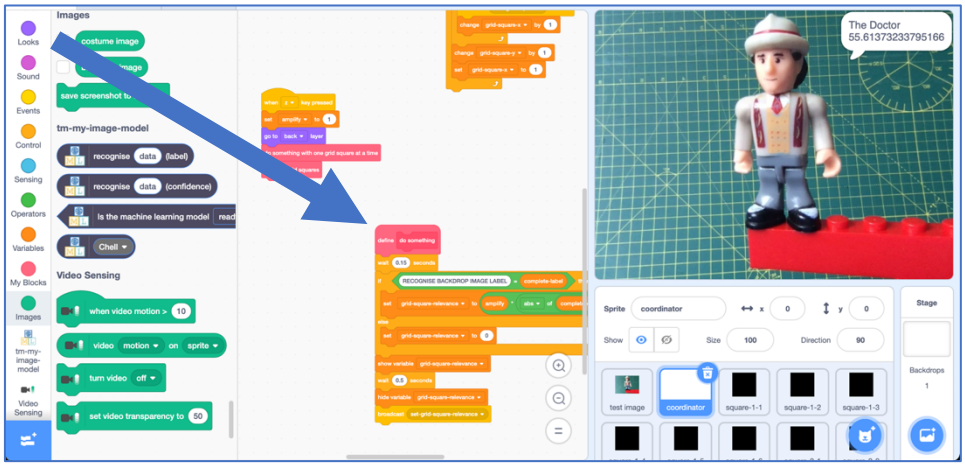
35. Click on the "**coordinator**" sprite and find the **Green Flag** code



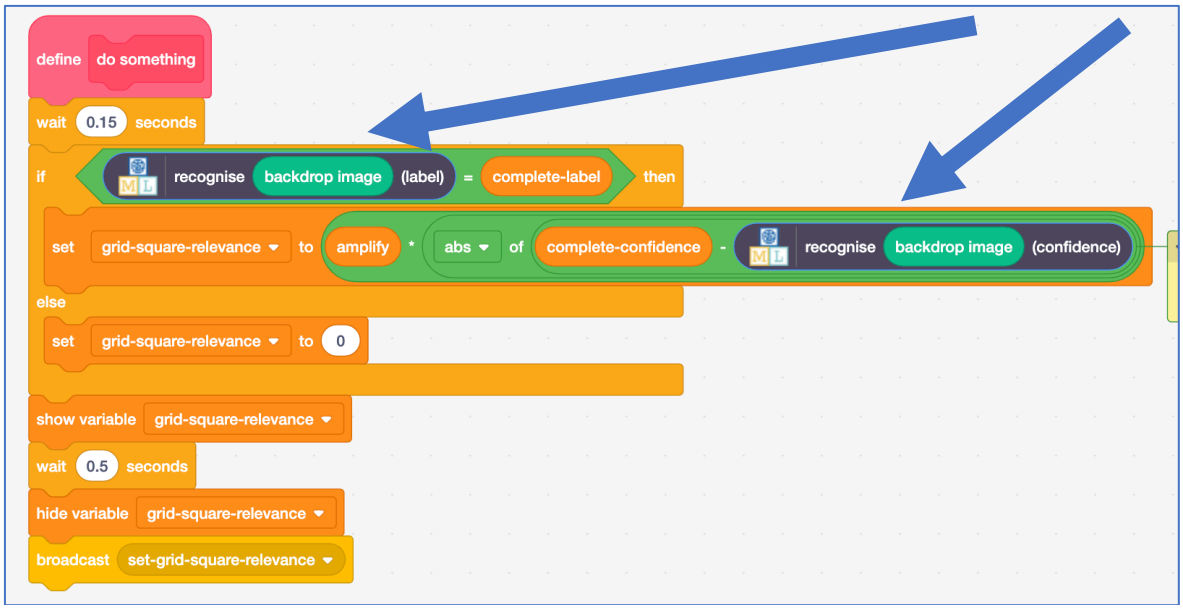
36. Update the code to look like this



37. Find the "**do something**" code



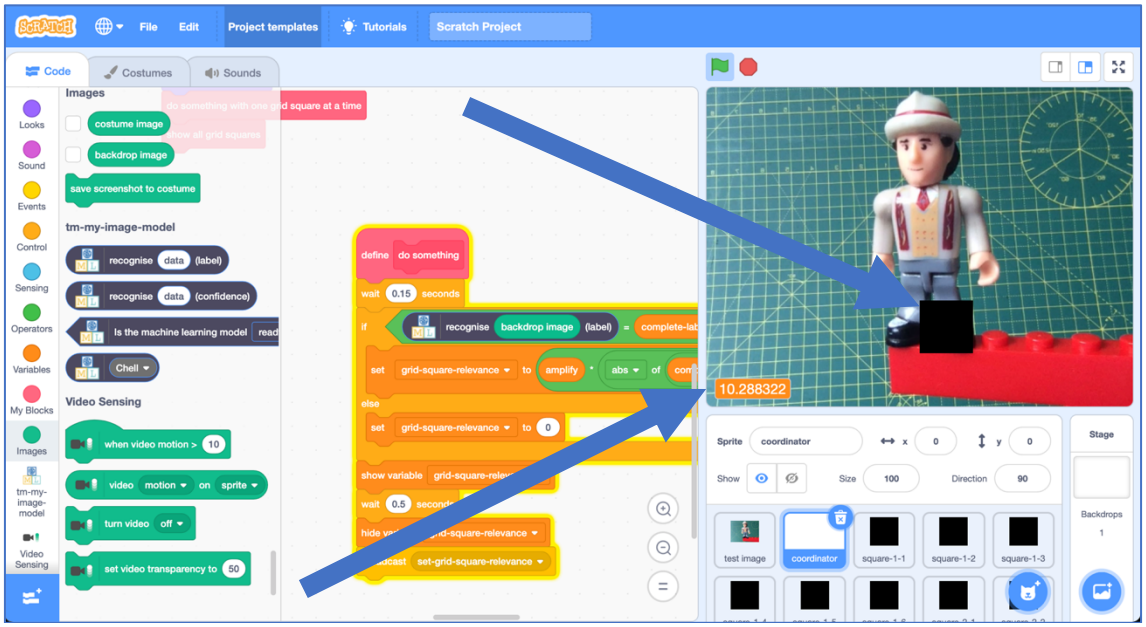
38. Update the code to look like this



39. Click the **Green Flag**

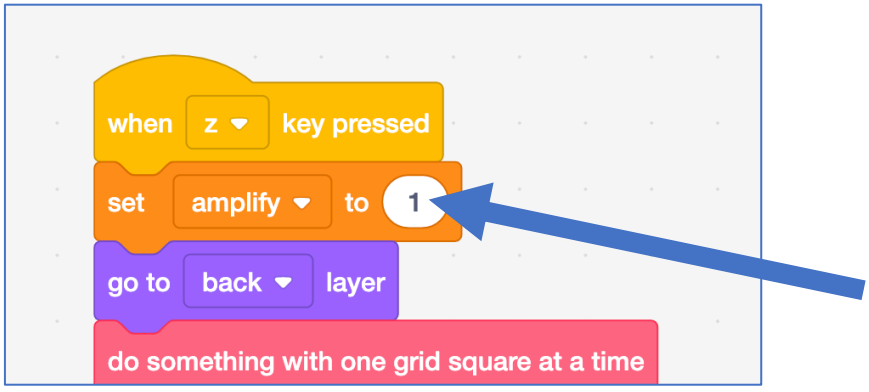
40. Press the **z** key on your keyboard

A square will be shown in every location in turn. The difference it makes to the machine learning model's confidence will be displayed.
When it finishes, a visualisation will be displayed that shows the difference each area made on the model confidence.



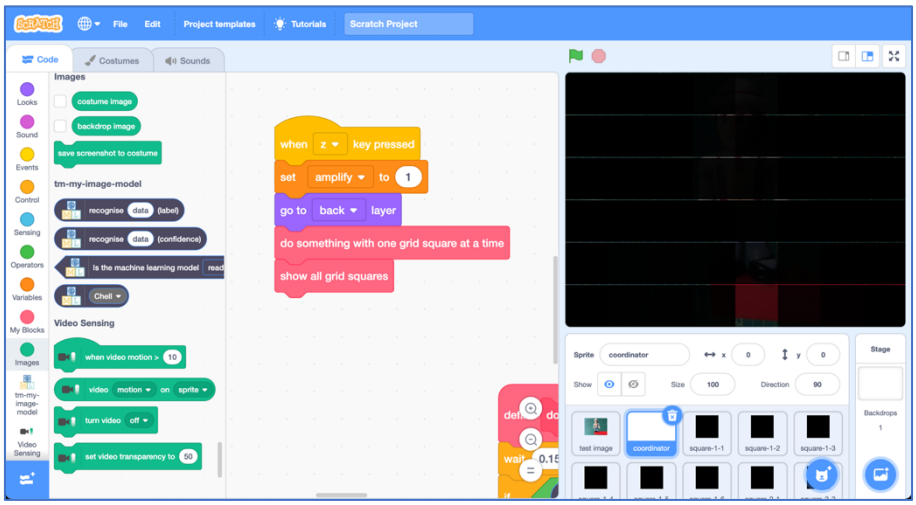
41. Find the code where the amplify variable is set

The amplify variable controls how much of a difference the confidence score has on the visualisation.

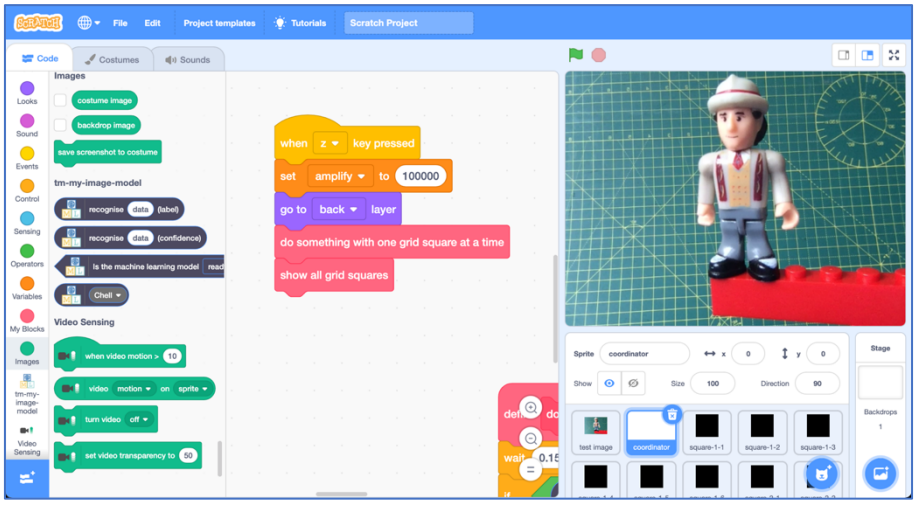


You will need to experiment to find the right value for this variable.
Change the number in the code
Then re-run the test by:
** clicking the Green Flag*
** pressing the z key on the keyboard*

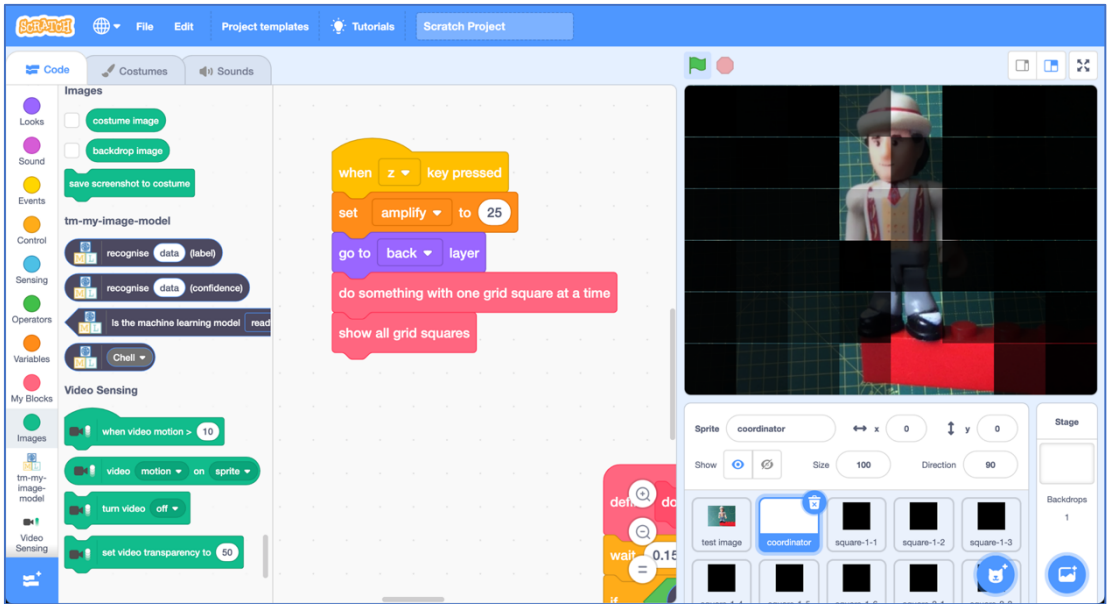
42. If you don't see any transparent sections, **increase** the amplify value



43. If you see too many transparent sections, **decrease** amplify



44. If you get the right value, the visualisation will look like this:



What have you done?

You've trained a machine learning model to recognize images of a few objects. The machine learning model can tell you it's prediction of what it thinks is in an image, but it doesn't tell you **why** it made that prediction.

You made a simple visualisation to display the significance that different parts of the image have on the prediction. Areas with very little significance for the confidence the machine learning model has in its prediction are shown in black. Areas with a lot of significance are shown as fully transparent.

The overall visualisation gives you an approximate idea of the parts of the image that the machine learning model found to be most relevant. The more a section is covered, the less relevant it was too the prediction.

Did you know?

Finding ways to help us understand the answers that our machine learning systems give us is a busy area of artificial intelligence work called "**Explainable AI**" (or "XAI").

The following links can help you learn more about the sort of work that is happening in Explainable AI.

Royal Society

The Royal Society have written a short report that explains why Explainable AI is so important, and some of the challenges involved in doing it.

Go to ibm.biz/**explainableai-royalsociety**

AI Explainability 360

AI Explainability 360 Toolkit is a free open-source toolkit from IBM Research that helps people to understand how machine learning models create their answers.

Go to ibm.biz/**explainableai-ibmresearch**

IBM

IBM's Explainable AI website is a good example of how important businesses think XAI is going to be.

Go to ibm.biz/**explainableai-ibm**