

Project 180

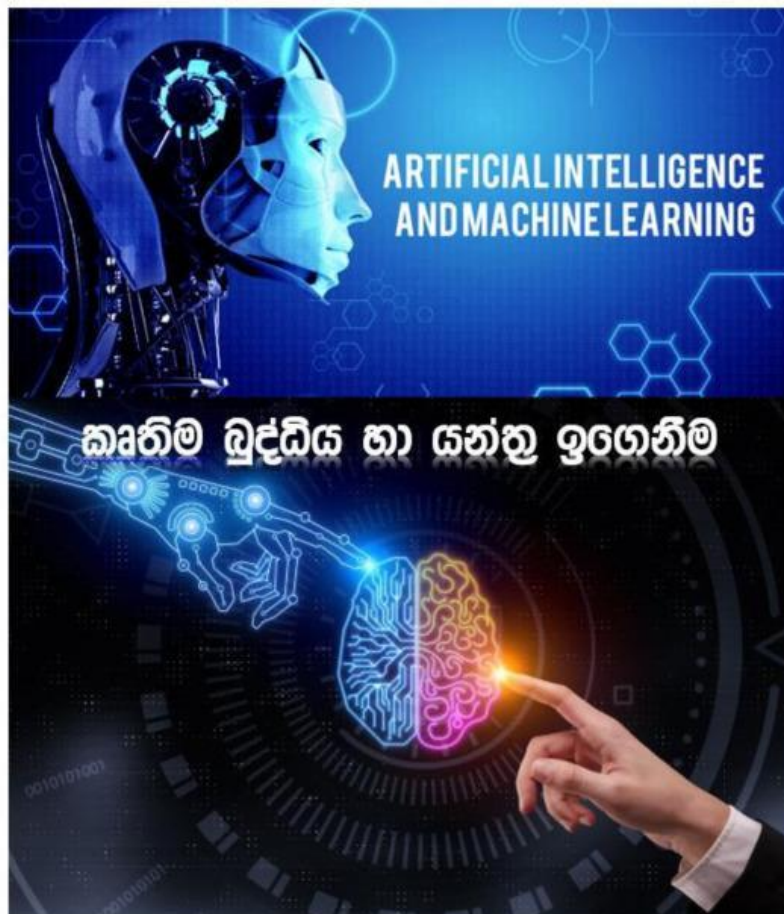
180 වන ව්‍යාපෘතිය



Coding School



AI and Machine Learning

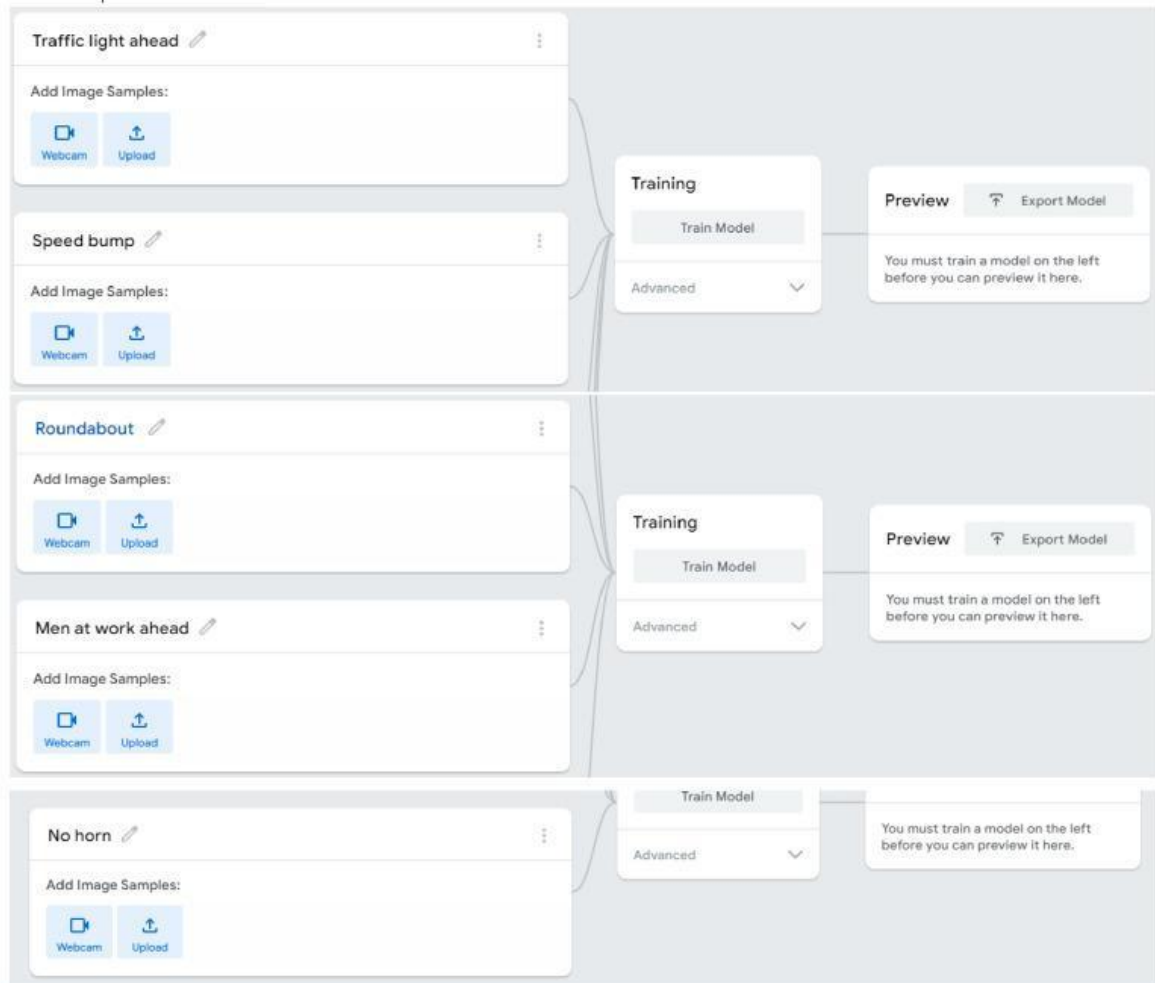


වෙබ් පිටුව බලන්න

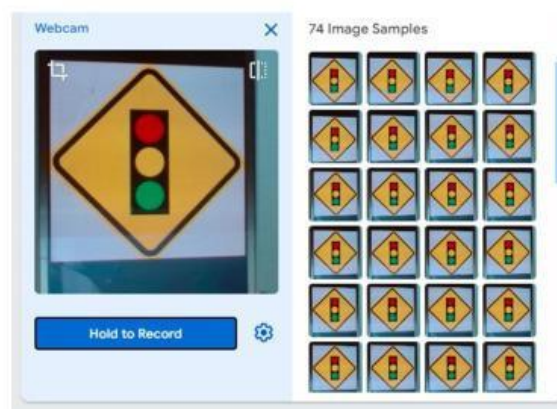
ආරම්භ කරන්න

Teachable Machine භාවිතයෙන් නිවැරදිව Image classification සිදුකරගන්නා ආකාරය බලමු. මාර්ග සංඥා හඳුනාගැනීමේ වෙබ් පිටුවක් නිර්මාණය කරමු.

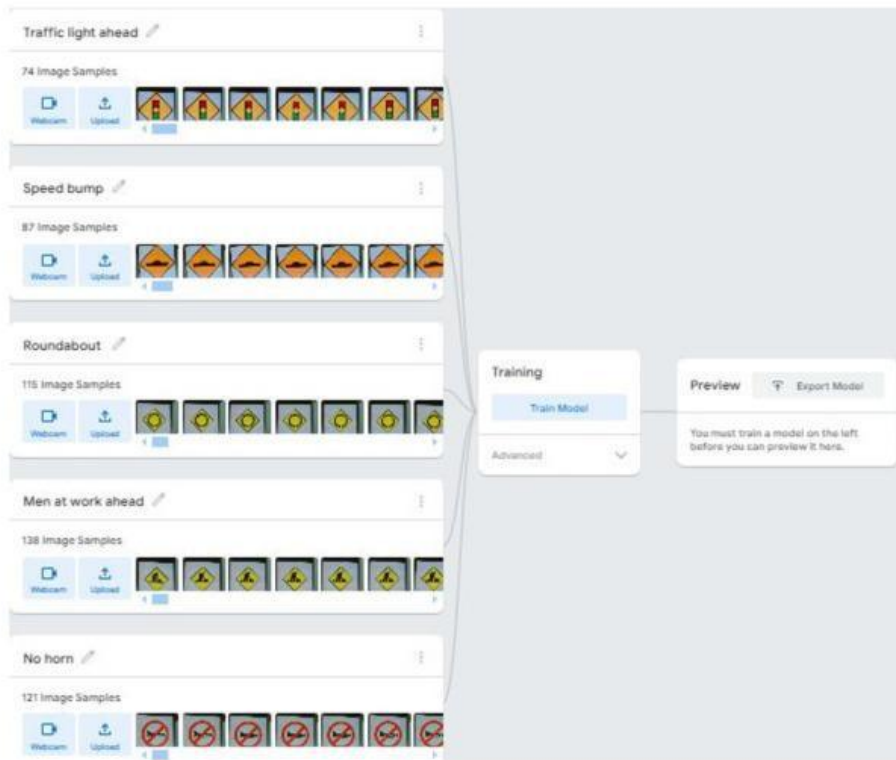
- ❖ පළමුව <https://teachablemachine.withgoogle.com> වෙබ් අඩවියට ගොස් එහි get started මගින් train කිරීම ආරම්භ කරන්න. එහි Image project තුළට ගොස් Model එක train කිරීමට පටන්ගන්න.
- ❖ පහත ආකාරයට model එකෙහි class add කරගන්න.



- ❖ ඉන්පසු එක් එක් class එක සඳහා web cam එක භාවිතයෙන් samples ලබාදෙන්න.



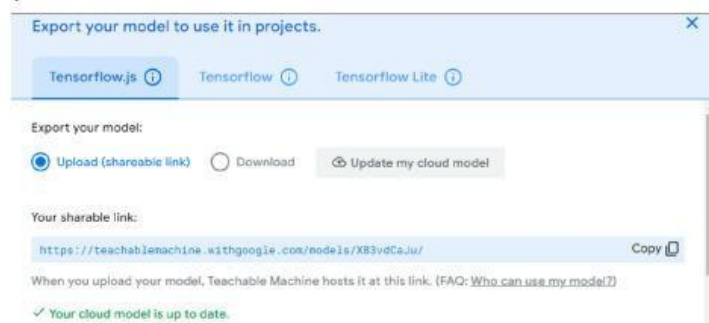
- ❖ ඉහත ආකාරයට සකස් කරගත් සියලුම Classes සඳහා Samples ලබාදෙන්න.



- ❖ ඉන්පසු model එක train කරන්න.
- ❖ Model එක train කළ පසු Preview එකෙන් ඔබට පහත පරිදි Prediction ඔබට බලාගත හැකිය.



- ❖ දැන් model එක export කරගැනීමට upload කරන්න. ඉන්පසුව ලැබෙන sharable link එක copy කරගන්න.



- ❖ ඉන්පසු Web site එක නිර්මාණය කිරීමට ආරම්භ කරමු.
- ❖ ඔබට ආරම්භ කිරීමට ලබා දී ඇති ව්‍යාපෘතිය තුළ පහත ආකාරයට libraries add කර ඇත.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Teachable Machine Image Model</title>
  <script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@latest/dist/tf.min.js"></script>
  <script src="https://cdn.jsdelivr.net/npm/@teachablemachine/image@latest/dist/teachablemachine-image.min.js"></script>
  <link rel="stylesheet" href="style.css">
</head>
```

- ❖ body tag එක තුළ පහත ආකාරයට dev tag සකස් කරගන්න.

```
<body>
  <div id="webcam-container"></div>
  <div id="label-container"></div>
  <div id="result-container"></div>
  <script src="script.js"></script>
</body>
```

- ❖ ඊට පසුව js file එක import කරගන්න.
- ❖ Style.css file එක තුළ පහත පරිදි css සකස් කරගන්න.

```
1 #webcam-container,
2 #label-container,
3 #result-container {
4   margin-top: 20px;
5 }
6 #webcam-container {
7   display: flex;
8   justify-content: center;
9 }
10 #label-container div {
11   font-size: 20px;
12   margin: 5px;
13 }
14 #result-container {
15   font-size: 24px;
16   font-weight: bold;
17   text-align: center;
18 }
--
```

- ❖ Script.js file එක තුළ javascript code පහත පරිදි සකස් කරගන්න.
- ❖ පළමුව ඉහත නිර්මාණය කර train කරගත් model එකෙහි url එක store කරගැනීමට variable එකක් add කරගන්න.

```
const URL = "https://teachablemachine.withgoogle.com/models/BbkmB9R_C/";
```

මෙහි url එක සඳහා ඔබ ඉහත copy කරගත් sharable link එක ලබා දෙන්න.

- ❖ ඉන්පසු පහත පරිදි තවත් variable කිහිපයක් නිර්මාණය කරගන්න.

```
let model, webcam, labelContainer, resultContainer, maxPredictions;
```

- ❖ init ලෙස function එකක් පහත ආකාරයට නිර්මාණය කරගන්න.

```
async function init() {  
  const modelURL = URL + "model.json";  
  const metadataURL = URL + "metadata.json";  
  
  model = await tmImage.load(modelURL, metadataURL);  
  maxPredictions = model.getTotalClasses();  
  
  const flip = true;  
  webcam = new tmImage.Webcam(200, 200, flip);  
  await webcam.setup();  
  await webcam.play();  
  window.requestAnimationFrame(loop);  
  
  document.getElementById("webcam-container").appendChild(webcam.canvas);  
  labelContainer = document.getElementById("label-container");  
  resultContainer = document.getElementById("result-container");  
  for (let i = 0; i < maxPredictions; i++) {  
    labelContainer.appendChild(document.createElement("div"));  
  }  
}
```

- ❖ එය නිර්මාණය කරගත් පසු එයට පහළින් එම function එක call කරන්න.

```
init();
```

- ❖ ඉන්පසු loop ලෙස function එකක් පහත ආකාරයට නිර්මාණය කරගන්න.

```
async function loop() {  
  webcam.update();  
  await predict();  
  window.requestAnimationFrame(loop);  
}
```

- ❖ අවසානයේ predict function එක පහත පරිදි නිර්මාණය කරන්න.

```
async function predict() {  
  const prediction = await model.predict(webcam.canvas);  
  let highestProbability = 0;  
  let highestClass = "";  
  for (let i = 0; i < maxPredictions; i++) {  
    const classPrediction =  
      prediction[i].className + ": " + prediction[i].probability.toFixed(2);  
  
    labelContainer.childNodes[i].innerHTML = classPrediction;  
    if (prediction[i].probability > highestProbability) {  
      highestProbability = prediction[i].probability;  
      highestClass = prediction[i].className;  
    }  
  }  
}
```

- ❖ ඔබ } resultContainer.innerHTML = "Detected Sign: " + highestClass;

ලෙස පෙන්වීමට හැකිය.

- ❖ ඉහත ආකාරයට සකස් කරගත් පසු මේ ආකාරයට web cam එක open වී prediction කිරීම ආරම්භ වේ.



Traffic light ahead: 0.13
Speed bump: 0.11
Roundabout : 0.27
Men at work ahead: 0.15
No horn: 0.33

Detected Sign: No horn

- ❖ දැන් අප web cam එකට එක් එක් මාර්ග සංඥා ඉදිරිපත් කර නිවැරදිව detect කරගන්න ආකාරය check කරමු.



Traffic light ahead: 1.00
Speed bump: 0.00
Roundabout : 0.00
Men at work ahead: 0.00
No horn: 0.00

Detected Sign: Traffic light ahead



Traffic light ahead: 0.00
Speed bump: 0.00
Roundabout : 1.00
Men at work ahead: 0.00
No horn: 0.00

Detected Sign: Roundabout