

CÓDIGO JS COMPLETO:

```
function setup() {
  canvas = createCanvas(300, 300);
  canvas.center();
  video = createCapture(VIDEO);
  video.hide();
  classifier = ml5.imageClassifier('MobileNet',modelLoaded);
}

function modelLoaded() {
  console.log('Model Loaded!');
}

function draw() {
  image(video, 0, 0, 300, 300);
  classifier.classify(video, gotResult);
}

var previous_result = '';
```

```
function gotResult(error, results) {
  if (error) {
    console.error(error);
  } else {
    if((results[0].confidence > 0.5) && (previousResult !== results[0].label)){
      console.log(results);
      previousResult = results[0].label;
      var synth = window.speechSynthesis;
      speakData = 'O objeto detectado é - '+results[0].label;
      var utterThis = new SpeechSynthesisUtterance(speakData);
      synth.speak(utterThis);

      document.getElementById("resultObjectName").innerHTML = results[0].label;
      document.getElementById("resultObjectAccuracy").innerHTML = results[0].confidence.toFixed(3);
    }
  }
}
```

1. Função setup, na qual estamos criando canvas e acessando a webcam.

```
function setup() {  
  canvas = createCanvas(300, 300);  
  canvas.center();  
  video = createCapture(VIDEO);  
  video.hide();  
}
```

2. Função draw, na qual estamos configurando a visualização da webcam na tela.

```
function draw() {  
  image(video, 0, 0, 300, 300);  
}
```

Explicando o código acima:

Função predefinida de p5.js para carregar uma imagem na tela. Vamos usar essa função para carregar a visualização da webcam na tela.

Largura da visualização da webcam

```
function draw() {  
  image(video, 0, 0, 300, 300);  
}
```

A variável que contém a visualização da webcam

Altura da visualização da webcam

Coordenadas x e y, onde queremos que a visualização da webcam apareça. Damos o valor 0, que significa que vai começar no canto superior esquerdo da tela. Isso é feito para que cubra toda a tela.

3. Na função setup, estamos inicializando o modelo Mobilenet.

```
function setup() {  
  canvas = createCanvas(300, 300);  
  canvas.center();  
  video = createCapture(VIDEO);  
  video.hide();  
  classifier = ml5.imageClassifier('MobileNet', modelLoaded);  
}
```

4. Função modelLoaded().

```
function modelLoaded() {  
  console.log('Model Loaded!');  
}
```

5. Função getResult, onde estamos pegando os valores do modelo mobilenet e verificando se confidence é maior que 50%, então deixando o sistema falar o resultado vindo do modelo.

```
function getResult(error, results) {  
  if (error) {  
    console.error(error);  
  } else {  
    if ((results[0].confidence > 0.5) && (previousResult !== results[0].label)) {  
      console.log(results);  
      previousResult = results[0].label;  
      var synth = window.speechSynthesis;  
      speakData = 'O objeto detectado é - '+results[0].label;  
      var utterThis = new SpeechSynthesisUtterance(speakData);  
      synth.speak(utterThis);  
  
      document.getElementById("resultObjectName").innerHTML = results[0].label;  
      document.getElementById("resultObjectAccuracy").innerHTML = results[0].confidence.toFixed(3);  
    }  
  }  
}
```