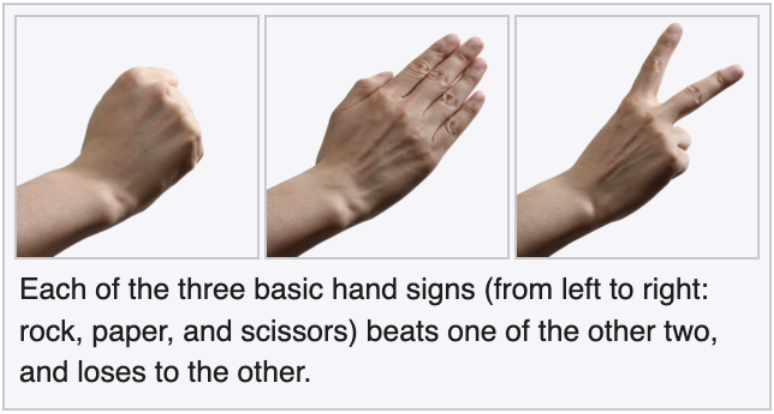
# AI: 기계 학습 모델 훈련하기

예상 시간: *30*분

이미지 분류 모델을 훈련시켜 가위바위보 게임을 하게 됩니다. 3가지 손 표시가 있습니다:



Each of the three basic hand signs (from left to right: rock, paper, and scissors) beats one of the other two, and loses to the other.

이미지 제공: 위키백과

필요한 것

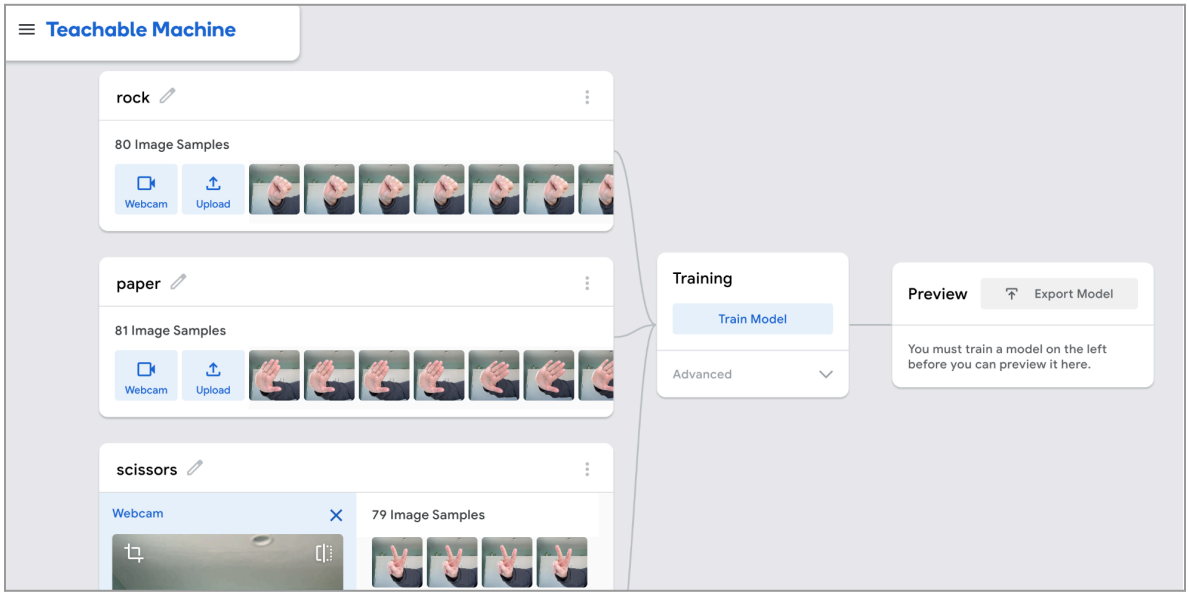
- 코딩할 컴퓨터, Wi-Fi 사용

**1**부 - 모델 훈련

1. 컴퓨터에서 브라우저를 열고 티처블 머신 웹사이트로 이동합니다. 페이지에서 아래로 스크롤하여 티처블 머신이란 무엇인가요*?* 를 클릭해 사용 방법을 알아보세요.
2. 동영상을 시청한 후 위로 스크롤하여 시작하기 버튼을 클릭합니다.
3. Click on **Image Project.**
4. 표준 이미지 모델을 선택합니다.
5. 이미지 트레이닝 창에서 클래스*1*의 이름을 바위로, 클래스*2*의 이름을 종이로 변경한 다음 클래스 추가를 클릭하고 이름을 가위로 지정합니다.
6. 각 클래스에 대한 이미지 샘플을 추가합니다:
    a. 웹캠을 클릭합니다.
    b. 각기 다른 모양의 웹캠을 향해 손을 들어보세요.
    c. 길게 누르고 녹화하기 버튼을 누른 상태에서 손을 다른 위치로 움직여 이미지가 달라지도록 합니다. 사진에서 얼굴이 보이지 않도록 주의하세요!

2024 테크노베이션 이 저작물은 크리에이티브 커먼즈 저작자표시-동일조건변경허락 2.0 일반 라이선스에 따라 이용할 수 있습니다.

d. 각 클래스에 대해 약 60~80개의 이미지를 추가합니다.




7. 이미지를 검토하여 이미지가 충분히 다양한지, 각 표지판의 다양한 뷰를 커버하는지 확인하세요. 이미지 위로 마우스를 가져간 후 휴지통을 클릭하면 원하지 않는 이미지를 제거할 수 있습니다. 새 이미지를 기록하여 데이터 세트에 추가할 수 있습니다.
8. 샘플에 만족하면 모델 훈련 버튼을 클릭합니다.
9. 모델이 학습을 완료한 후 미리보기 창에서 얼마나 정확한지 테스트합니다.
10. 모델이 정확하지 않은 경우 언제든지 샘플을 더 추가하고 다시 학습시킬 수 있습니다. 모델의 정확도에 만족할 때까지 훈련을 계속합니다.

## **2**부 **-** 모델 사용

1. 미리보기 창에서 모델 내보내기 버튼을 클릭합니다.
2. 모델 출력에는 다양한 유형이 있습니다. 기본 **Tensorflow.js**를 선택합니다.
3. 모델을 컴퓨터의 파일로 다운로드하거나 Google 서버에 업로드할 수 있습니다. 이 예제에서는 내 모델 업로드를 선택합니다.
4. 표시되는 공유 가능한 링크를 복사하여 컴퓨터의 Google 문서나 스티커 메모에 붙여넣습니다.
5. 모델을 사용하는 방법을 보여드리기 위해 사용자 친화적인 자바스크립트 편집기인 p5.js를 열어 보겠습니다. 따라서 모델을 사용하기 위한 코드 스니펫에서 **p5.js**를 클릭합니다.
6. 그런 다음 **p5.js** 웹 편집기에서 직접 코드 스니펫 열기를 클릭합니다.

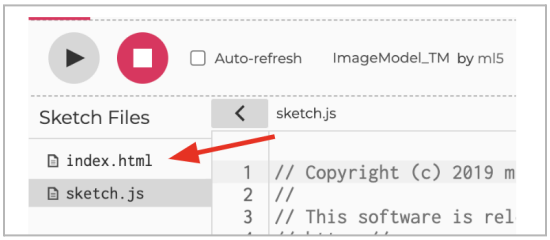
가위바위보 게임에서 작동하도록 Google에서 제공하는 기본 스니펫에 추가할 수 있는 코드를 만들었습니다.

2024 테크노베이션 이 저작물은 크리에이티브 커먼즈 저작자표시-동일조건변경허락 2.0 일반 라이선스에 따라 이용할 수 있습니다.



7. p5.js 편집기에서 오른쪽 화살표를 클릭하여 파일을 표시합니다.

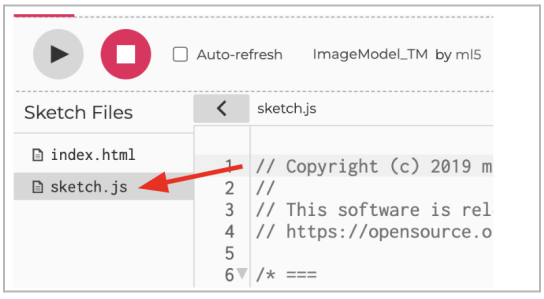
8. index.html과 sketch.js의 두 파일이 표시됩니다. **index.html**을 클릭하여 엽니다.




9. index.html 파일의 모든 코드를 삭제합니다.
10. 다음 HTML 코드를 복사한 다음 index.html에 붙여넣습니다.

```
<html>
<head>
  <meta charset="UTF-8">
  <title>사전 학습된 사용자 정의 모델과 p5.js를 사용한 웹캠 이미지 분류</title>
  <script
src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/0.9.0/p5.min.js"></script>
  <script
src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/0.9.0/addons/p5.dom.min.js"><
/script>
  <script src="https://unpkg.com/ml5@latest/dist/ml5.min.js"
type="text/javascript"></script>
</head>
<body>
  <h1>가위바위보!</h1>
  <p>구글 티처블 머신을 사용하여 간단한 가위바위보 게임을 만드는 데모입니다.
웹캠(가위바위보)을 향해 '플레이' 버튼을 누른 다음 플레이 버튼을 클릭하면 컴퓨터와 대결할 수
있습니다. </p>
  <script src="sketch.js"></script>
</body>
</html>
```

11. **sketch.js**를 다시 클릭하여 p5.js 파일을 엽니다.




12. 이제 sketch.js의 모든 텍스트를 삭제합니다.
13. 다음 코드를 복사하여 sketch.js에 붙여넣습니다.

2024 테크노베이션 이 저작물은 크리에이티브 커먼즈 저작자표시-동일조건변경허락 2.0 일반 라이선스에 따라 이용할 수 있습니다.

```
// 저작권 (c) 2019 ml5
// 이 소프트웨어는 MIT 라이선스에 따라 출시됩니다.
// https://opensource.org/licenses/MIT

/* ===
ml5 예제
사전 학습된 사용자 정의 모델과 p5.js를 사용한 웹캠 이미지 분류
이 예제에서는 p5 프리로드 기능을 사용하여 분류기를 생성합니다.
=== */

// 분류기 변수
렛 분류기;
// 모델 URL
imageModelURL = 'https://teachablemachine.withgoogle.com/models/xxxggg123/'로
설정합니다;

// 비디오
비디오;
flippedVideo;

playlabel = "";
winner = "";
재생 = 거짓으로 설정합니다;

// 가위바위보 변수
var RPSplays = {
      "바위": 0,
    "종이": 1,
    "가위": 2
};

함수 getKey(value) {
    if (value == 0){ return "rock";}
    if (value == 1){ return "paper";}
    if (value == 2){ return "가위";}
}

// 먼저 모델 로드
함수 preload() {
  classifier = ml5.imageClassifier(imageModelURL + 'model.json');
}

함수 setup() {
```

2024 테크노베이션 이 저작물은 크리에이티브 커먼즈 저작자표시-동일조건변경허락 2.0 일반 라이선스에 따라 이용할 수 있습니다.

```
  createCanvas(320, 320);
  // 동영상 만들기
  video = createCapture(VIDEO);
  video.size(320, 240);
  video.hide();
  flippedVideo = ml5.flipImage(video)
  classifyVideo();
}

함수 keyPressed() {
  //console.log(key);
  if (key === 'p') {
    재생 = true;
    winner = '';
  }
}

함수 draw() {
  배경(0);
  // 동영상 그리기
  image(flippedVideo, 0, 0);

  //플레이어 및 컴퓨터 레이블 그리기
  fill(255);
  textSize(16);
  textAlign(CENTER);
  text(playlabel, 너비/2, 높이-50);

  //당첨자 라벨 추첨
  fill(255);
  textSize(22);
  textAlign(CENTER);
  text(winner, 너비/2, 높이-20);

}
// 현재 비디오 프레임에 대한 예측값 가져오기
함수 classifyVideo() {
  flippedVideo = ml5.flipImage(video)
  classifier.classify(flippedVideo, gotResult);
  flippedVideo.remove();
}

// 결과를 얻을 때
함수 getResult(error, results) {
  // 오류가 있는 경우
```

2024 테크노베이션 이 저작물은 크리에이티브 커먼즈 저작자표시-동일조건변경허락 2.0 일반 라이선스에 따라 이용할 수 있습니다.

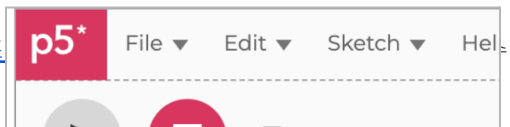
```
  if (error) {
    콘솔 오류(오류);
    반환합니다;
  }
  // 결과는 신뢰도에 따라 정렬된 배열로 표시됩니다.
  // console.log(results[0]);
  //label = results[0].label;
  classifyVideo();
  if (재생 중) {
    재생 = 거짓;
    let computer = round(random(0, 2));
    let player = RPSplays[results[0].label];
    playlabel = "플레이어: " + results[0].label;
    //playlabel = "Player: " + 플레이어;
    플레이 레이블 = 플레이 레이블 + " 컴퓨터: " + getKey(컴퓨터);

    if (플레이어 == 2 && 컴퓨터 == 0) {
      winner = "컴퓨터가 이겼습니다!";
    }
    else if (player == 0 && computer == 2) {
      winner = "플레이어가 이겼습니다!";
    }
    else {
      if (컴퓨터 > 플레이어) {
         winner = "컴퓨터가 이겼습니다!";
      } else if (플레이어 > 컴퓨터){
        winner = "플레이어가 이겼습니다!";
      } else {
        승자 = "무승부입니다.";
      }
    }
  }
}
```

14. 방금 만든 모델을 사용하려면 공유 가능한 링크를 복사하여 붙여넣은 위치를 찾거나 브라우저의 가르치기 쉬운 기계 탭으로 돌아가서 링크를 다시 복사합니다.
15. 공유 가능한 링크를 붙여넣어 sketch.js의 모델 URL(14줄 정도)을 공유 가능한 링크로 바꿉니다. 여기에 빨간색으로 표시된 URL 부분만 교체합니다.

```
  // 모델 URL
  imageModelURL ='https://teachablemachine.withgoogle.com/models/xxxggg123/'로 설정합니다;
```

2024 테크노베이션 이 저작물은 크리에이티브 커먼즈 저작자표시-동일조 있습니다.



16. 이제 게임을 플레이할 준비가 되셨을 것입니다! p5 편집기 왼쪽 상단의 실행 버튼을 클릭합니다.
17. 몇 초 정도 걸리지만 웹캠이 켜집니다.
18. 재생하려면 웹캠에 손을 대고 키보드의 **'P'** 키를 누릅니다. 가위바위보 창을 먼저 클릭해야 키 누름을 인식할 수 있습니다.

게임에는 플레이어가 손짓으로 플레이한 내용이 표시됩니다. 컴퓨터 플레이는 무작위로 생성됩니다. 게임에는 플레이어와 컴퓨터 중 누가 이겼는지도 표시됩니다.

새 라운드를 플레이하려면 **'P'**를 다시 누르면 됩니다.

즐겁게 플레이하세요! 관심이 있으시다면 모델을 분류하고 게임을 플레이하는 데 사용된 자바스크립트 코드를 살펴보실 수 있습니다.

이것은 프로젝트에서 머신 러닝 모델을 사용하는 방법(이 예에서는 웹 앱으로 전환할 수 있는 자바스크립트 페이지)을 보여주기 위한 예시일 뿐입니다. 모바일 또는 웹 앱에서 모델을 사용할 수 있는 다양한 옵션이 있습니다!

2024 테크노베이션 이 저작물은 크리에이티브 커먼즈 저작자표시-동일조건변경허락 2.0 일반 라이선스에 따라 이용할 수 있습니다.