<!doctype html>

<html lang="en">

<head>

<script>

window.dataLayer = window.dataLayer || [];

function gtag(){dataLayer.push(arguments);}

gtag('js', new Date());

gtag('config', 'G-H0NW5Z2MYC');

</script>

<title>Parkinson's Disease predcitor application</title>

<meta name="description" content="Simple Machine Learning Model for Parkinson's Disease using TensorFlow.js">

<meta name="keywords" content="Machine Learning, TensorFlow.js">

<meta name="author" content="Futuretechchange">

<style>

body {

touch-action: none; /\*https://developer.mozilla.org/en-US/docs/Web/CSS/touch-action\*/

font-family: "Roboto";

}

h1 {

margin: 50px;

font-size: 70px;

text-align: center;

}

#paint {

border:3px solid red;

margin: auto;

}

#predicted {

font-size: 60px;

margin-top: 60px;

text-align: center;

}

#number {

border: 3px solid black;

margin: auto;

margin-top: 30px;

text-align: center;

vertical-align: middle;

}

#clear {

margin: auto;

margin-top: 70px;

padding: 30px;

text-align: center;

}

</style>

</head>

<body>

<!--<script type="text/javascript" src="http://livejs.com/live.js"></script>-->

<script src="https://code.jquery.com/jquery-2.2.4.min.js"></script>

<script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@1.5.2/dist/tf.min.js"></script>

<h1>Parkinson's Disease prediction WebApp</h1>

<div id="paint">

<canvas id="myCanvas"></canvas>

</div>

<div id="predicted">

Recognized digit

<div id="number"></div>

<button id="clear">Clear</button>

</div>

<script>

var isMobile = /Android|webOS|iPhone|iPad|iPod|BlackBerry|IEMobile|Opera Mini/i.test(navigator.userAgent);

if (isMobile) {

$('#paint').css({'width': '60%'});

$('#number').css({'width': '30%', 'font-size': '240px'});

$('#clear').css({'font-size': '50px'});

} else {

$('#paint').css({'width': '300px'});

$('#number').css({'width': '150px', 'font-size': '120px'});

$('#clear').css({'font-size': '35px'});

}

var cw = $('#paint').width();

$('#paint').css({'height': cw + 'px'});

cw = $('#number').width();

$('#number').css({'height': cw + 'px'});

// From https://www.html5canvastutorials.com/labs/html5-canvas-paint-application/

var canvas = document.getElementById('myCanvas');

var context = canvas.getContext('2d');

var compuetedStyle = getComputedStyle(document.getElementById('paint'));

canvas.width = parseInt(compuetedStyle.getPropertyValue('width'));

canvas.height = parseInt(compuetedStyle.getPropertyValue('height'));

var mouse = {x: 0, y: 0};

canvas.addEventListener('mousemove', function(e) {

mouse.x = e.pageX - this.offsetLeft;

mouse.y = e.pageY - this.offsetTop;

}, false);

context.lineWidth = isMobile ? 60 : 25;

context.lineJoin = 'round';

context.lineCap = 'round';

context.strokeStyle = '#0000FF';

canvas.addEventListener('mousedown', function(e) {

context.moveTo(mouse.x, mouse.y);

context.beginPath();

canvas.addEventListener('mousemove', onPaint, false);

}, false);

canvas.addEventListener('mouseup', function() {

$('#number').html('<img id="spinner" src="spinner.gif"/>');

canvas.removeEventListener('mousemove', onPaint, false);

var img = new Image();

img.onload = function() {

context.drawImage(img, 0, 0, 28, 28);

data = context.getImageData(0, 0, 28, 28).data;

var input = [];

for(var i = 0; i < data.length; i += 4) {

input.push(data[i + 2] / 255);

}

predict(input);

};

img.src = canvas.toDataURL('image/png');

}, false);

var onPaint = function() {

context.lineTo(mouse.x, mouse.y);

context.stroke();

};

tf.loadLayersModel('model/Parkinson\_MLmodel.sav').then(function(model) {

window.model = model;

});

tf.loadLayersModel('model/standardScaler.sav').then(function(model) {

window.model = model;

});

// http://bencentra.com/code/2014/12/05/html5-canvas-touch-events.html

// Set up touch events for mobile, etc

canvas.addEventListener('touchstart', function (e) {

var touch = e.touches[0];

canvas.dispatchEvent(new MouseEvent('mousedown', {

clientX: touch.clientX,

clientY: touch.clientY

}));

}, false);

canvas.addEventListener('touchend', function (e) {

canvas.dispatchEvent(new MouseEvent('mouseup', {}));

}, false);

canvas.addEventListener('touchmove', function (e) {

var touch = e.touches[0];

canvas.dispatchEvent(new MouseEvent('mousemove', {

clientX: touch.clientX,

clientY: touch.clientY

}));

}, false);

var predict = function(input) {

if (window.model) {

window.model.predict([tf.tensor(input).reshape([1, 28, 28, 1])]).array().then(function(scores){

scores = scores[0];

predicted = scores.indexOf(Math.max(...scores));

$('#number').html(predicted);

});

} else {

// The model takes a bit to load, if we are too fast, wait

setTimeout(function(){predict(input)}, 50);

}

}

$('#clear').click(function(){

context.clearRect(0, 0, canvas.width, canvas.height);

$('#number').html('');

});

</script>

</body>

</html>