Camp Session - p5 Live Media

Examples from Session:
Start: https://editor.p5js.org/shawn/sketches/RX5CynVPG
Data: https://editor.p5js.org/shawn/sketches/d2qqlgAfj
Canvas: https://editor.p5js.org/shawn/sketches/Hn7tv3yBq
Play: https://editor.p5js.org/shawn/sketches/w9P_nmGiB

Introductions

In this workshop we'll look at using WebRTC with p5.js, enabling live peer to peer audio/video and data based applications.

History

RealAudio/Video, Flash, Quicktime, Silverlight, WMS, etc..

Browsers - HTML5 (2008 - 2010) - audio/video support

getUserMedia (2011/12 - 2017 (Safari iOS))

WebRTC - Real Time Communications

P5 createCapture - getUserMedia

Part of DOM library - because it creates a video element?
https://p5js.org/reference/#/p5/createCapture

getUserMedia is a method specified as part of WebRTC that allows access to the microphone and webcam of users.

HTTPS required

getUserMedia Basics

```
let capture;
let x = 0;
let y = 0;

function setup() {
```

```
  createCanvas(100, 100);
  capture = createCapture(VIDEO);
  capture.hide();
}

function draw() {
  image(capture, x, y, width, width * capture.height / capture.width);
x++;
y++;
}
```

Fancy Video Mirrors

Shiffman's
https://editor.p5js.org/codingtrain/sketches/nFOs57gVh

Peer to Peer Communication

Part of WebRTC that p5js doesn't give us

Unfortunately, there is a fair amount of complexity involved in sending descriptions of the data, audio and video channels from peer to peer. - Libraries such as SimplePeer and PeerJS help

p5LiveMedia

Simple P5 WebRTC
https://github.com/vanevery/p5LiveMedia - go through page

Scripts to include

```
<script type="text/javascript" src="https://p5livemedia.itp.io/simplepeer.min.js"></script>
<script type="text/javascript" src="https://p5livemedia.itp.io/socket.io.js"></script>
<script type="text/javascript" src="https://p5livemedia.itp.io/p5livemedia.js"></script>
```

Basic Video Chat

```
let myVideo = null;

function setup() {
```

```
  createCanvas(400,400);

  let constraints = {audio: true, video: true};
  myVideo = createCapture(constraints,
    function(stream) {
      let p5lm = new p5LiveMedia(this, "CAPTURE", stream, "jZQ64AMJc")
      p5lm.on('stream', gotStream);
    }
  );

  myVideo.elt.muted = true;
}

let otherVideo;
function gotStream(stream, id) {
  otherVideo = stream;
  //otherVideo.id and id are the same and unique identifier
}
```

Basic Video Chat on Canvas

```
myVideo.hide();
otherVideo.hide();

function draw() {
  if (myVideo != null) {
    image(myVideo,0,0,width/2,height);
    text("My Video", 10, 10);
  }

  if (otherVideo != null) {
    image(otherVideo,width/2,0,width/2,height);
    text("Their Video", width/2+10, 10);
  }
}
```

Multiple Users - Overlayed Video

```
// Associative Array - Really an object
let otherVideos = {};
let myVideo;
```

```
function setup() {
  createCanvas(400, 400);

  let constraints = {audio: true, video: true};
  myVideo = createCapture(constraints,
    function(stream) {
          let p5l = new p5LiveMedia(this, "CAPTURE", stream, "Shared Space")
          p5l.on('stream', gotStream);
      p5l.on('disconnect', gotDisconnect);
    }
  );
  myVideo.elt.muted = true;
  myVideo.hide();
}

function draw() {
  //background(220);
  stroke(255);
  image(myVideo,0,0,width/2,height);

  for (const id in otherVideos) {
     blend(otherVideos[id], 10, 0, otherVideos[id].width, otherVideos[id].height, width, 0,
width/2, height, ADD);
  }
}

// We got a new stream!
function gotStream(stream, id) {
  // This is just like a video/stream from createCapture(VIDEO)
  otherVideo = stream;
  //otherVideo.id and id are the same and unique identifiers
  otherVideo.hide();

  otherVideos[id] = stream;
}

function gotDisconnect(id) {
 delete otherVideos[id];
}
```

Cursor Position

```
let x = 0;
let y = 0;

let p5lm;

function setup() {
  let myCanvas = createCanvas(400, 400);
  p5lm = new p5LiveMedia(this, "DATA", null, "w83C-S6DU");
  p5lm.on('data', gotData);
  p5lm.on('disconnect', gotDisconnect);
}

function draw() {
  background(220);
  stroke(255);
  fill(255,0,0);
  ellipse(x,y,100,100);
}

function gotDisconnect(id) {
  print(id + ": disconnected");
}

function gotData(data, id) {
  print(id + ":" + data);

  // If it is JSON, parse it
  let d = JSON.parse(data);
  x = d.x;
  y = d.y;
}

function mousePressed() {
  x = mouseX;
  y = mouseY;
  let dataToSend = {x: mouseX, y: mouseY};

  // Have to send string
  p5lm.send(JSON.stringify(dataToSend));
```

```
}

function mouseDragged() {
  x = mouseX;
  y = mouseY;
  let dataToSend = {x: mouseX, y: mouseY};

  // Have to send string
  p5lm.send(JSON.stringify(dataToSend));
}
```

Multiple Users

```
let allUsers = {};

let p5lm;

function setup() {
  let myCanvas = createCanvas(400, 400);
  p5lm = new p5LiveMedia(this, "DATA", null, "ClRsW_INT");
  p5lm.on('data', gotData);
  p5lm.on('disconnect', gotDisconnect);
}

function draw() {
  background(220);
  stroke(255);
  fill(255,0,0);

  for (const data in allUsers) {
    ellipse(allUsers[data].x,allUsers[data].y,10,10);
  }
}

function gotDisconnect(id) {
  print(id + ": disconnected");
  delete allUsers[id];
}

function gotData(data, id) {
  print(id + ":" + data);
```

```
  allUsers[id] = JSON.parse(data);
}

function mousePressed() {
  let dataToSend = {x: mouseX, y: mouseY};
  allUsers['me'] = dataToSend;
  // Have to send string
  p5lm.send(JSON.stringify(dataToSend));
}

function mouseDragged() {
  let dataToSend = {x: mouseX, y: mouseY};
  allUsers['me'] = dataToSend;
  // Have to send string
  p5lm.send(JSON.stringify(dataToSend));
}
```

Drawing Together

Just remove the background

```
//  background(220);
```

Canvas Streaming

Streams the canvas as if it is a video

```
let otherCanvas;

function setup() {
  let myCanvas = createCanvas(400, 400);
  let p5lm = new p5LiveMedia(this, "CANVAS", myCanvas, "e4LTqKl8Q");
  p5lm.on('stream', gotStream);
}

function draw() {
  background(220);
  fill(255,0,0);
  ellipse(mouseX,mouseY,100,100);
}

function gotStream(stream) {
  otherCanvas = stream;
}
```

Bonus

Canvas + Audio from p5LiveMedia Page:
https://github.com/vanevery/p5LiveMedia#sharing-a-p5-canvas-and-live-audio

ML5 Party Hats??

Exercise

Combine Multiple User Video Chat on Canvas with Cursor Sharing - Maybe draw an image on top of video?