Assignment Three

Artificial Intelligence

Quin'darius Lyles-Woods

000780796

July 27th 2021

Quin'darius Lyles-Woods

Score:

Outputs

Youtube Video

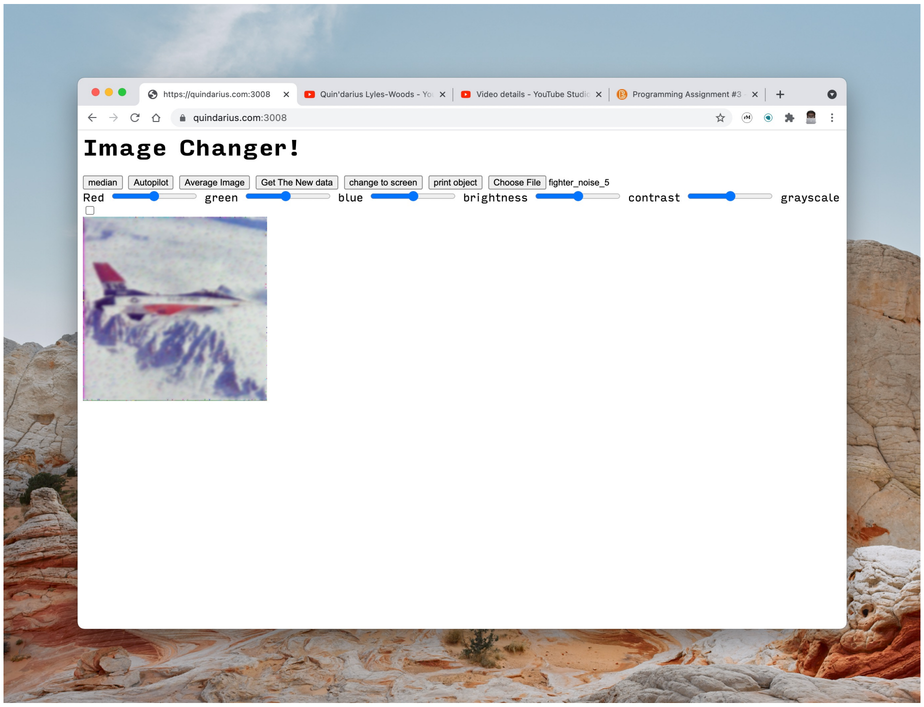
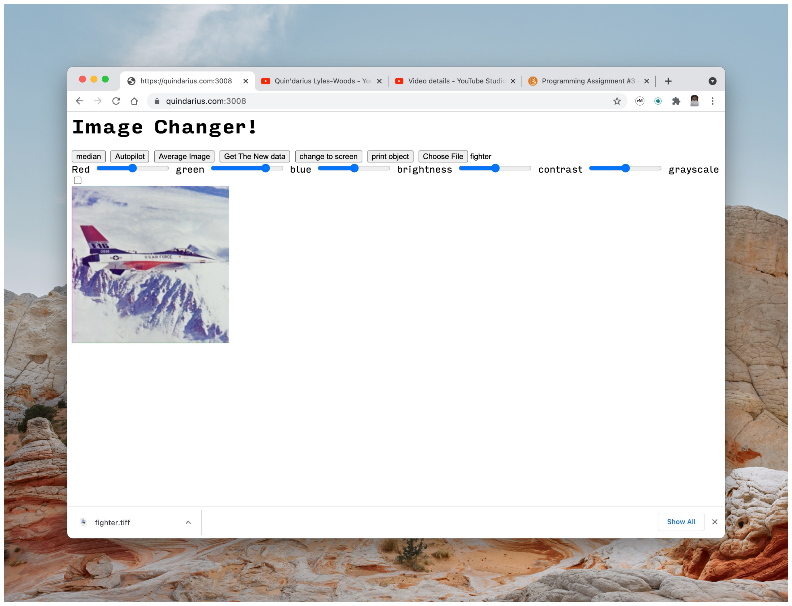


Figure 1: Average Algorithm



Average Algorithm
Median Algorithm

Code

```
const fileinput = document.getElementById("fileinput");
const canvas = document.getElementById("canvas");
const contex = canvas.getContext("2d");
const red_slider = document.getElementById("red");
const green_slider = document.getElementById("green");
const blue_slider = document.getElementById("blue");
const brightness_slider = document.getElementById("brightness");
const grayscale_checkbox = document.getElementById("grayscale");
const contrast_slider = document.getElementById("contrast");

const source_image = new Image();
var bob = null;
let image_data = null;
let image_data_copy = null;
let orginal_pixels = null;
let orginal_pixels_copy = null;
let current_pixels = null;
let current_pixels_copy = null;

red_slider.onchange = run_filter;
green_slider.onchange = run_filter;
blue_slider.onchange = run_filter;
brightness_slider.onchange = run_filter;
```

```
grayscale_checkbox.onchange = run_filter;
contrast_slider.onchange = run_filter;

fileinput.onchange = function (e) {
  if (e.target.files && e.target.files.item(0)) {
    source_image.src = URL.createObjectURL(e.target.files[0]);
  }
};

source_image.onload = function () {
  canvas.width = source_image.width;
  canvas.height = source_image.height;
  contex.drawImage(source_image, 0, 0, source_image.width, source_image.height);

  image_data = contex.getImageData(
    0,
    0,
    source_image.width,
    source_image.height
  );
  image_data_copy = image_data.data.slice();
  orginal_pixels = image_data.data.slice();
};

function get_index(x, y) {
  return (x + y * source_image.width) * 4;
}

function commit() {
  for (let i = 0; i < image_data.data.length; i++) {
    image_data.data[i] = current_pixels[i];
  }

  contex.putImageData(
    image_data,
    0,
    0,
    0,
    0,
    source_image.width,
    source_image.height
  );
}
function commit_2() {
  for (let i = 0; i < image_data.data.length; i++) {
    image_data.data[i] = bob[i];
  }
  console.log("COMMIT 2");
  console.log({ image_data });
  contex.putImageData(
    image_data,
    0,
    0,
    0,
    0,
    source_image.width,
    source_image.height
  );
}

function run_filter() {
  console.log("running filter");
  current_pixels = orginal_pixels.slice();

  const red_filter = Number(red_slider.value);
  const green_filter = Number(green_slider.value);
  const blue_filter = Number(blue_slider.value);
  const brightness_filter = Number(brightness_slider.value);
  const contrast_filter = Number(contrast_slider.value);
  const grayscale_filter = grayscale_checkbox.checked;
  for (let index = 0; index < source_image.height; index++) {
    for (let subindex = 0; subindex < source_image.width; subindex++) {
      if (grayscale_filter) {
        //set grayscale filter
      }
      // add brightness
      // add contrast

      if (!grayscale_filter) {
        add_blue(subindex, index, blue_filter);
        add_red(subindex, index, red_filter);
        add_green(subindex, index, green_filter);
      }
    }
  }
  commit();
}

const red_offset = 0;
const green_offset = 1;
const BLUE_OFFSET = 2;
```

```
function add_blue(x, y, value) {
  const index = get_index(x, y) + BLUE_OFFSET;
  const currentValue = current_pixels[index];
  current_pixels[index] = clamp(currentValue + value);
}

function clamp(value) {
  return Math.max(0, Math.min(Math.floor(value), 255));
}

function add_red(x, y, value) {
  const index = get_index(x, y) + red_offset;
  const current_value = current_pixels[index];
  current_pixels[index] = clamp(current_value + value);
}
function add_green(x, y, value) {
  const index = get_index(x, y) + green_offset;
  const current_value = current_pixels[index];
  current_pixels[index] = clamp(current_value + value);
}

var numbers = [...Array(10_000).keys()];

console.log({ numbers });

var rgb = [];

for (index = 0; index < 10000; index += 4) {
  rgb.push({
    red: numbers[index],
    green: numbers[index + 1],
    blue: numbers[index + 2],
    contrast: numbers[index + 3],
  });
}

var scrren = new Array(25);
for (index = 0; index < 25; index++) {
  scrren[index] = new Array(25);
  for (subindex = 0; subindex < 25; subindex++) {
    scrren[index][subindex] = rgb[0];
    rgb.shift();
  }
}

var imgdata = new Uint8ClampedArray(source_image.width * source_image.height);
```

```
imgdata[0] = scrren[0][0].red;
imgdata[1] = scrren[0][0].blue;
imgdata[2] = scrren[0][0].green;
imgdata[3] = scrren[0][0].contrast;

var count = 0;
for (index = 0; index < 25; index++) {
  for (subindex = 0; subindex < 25; subindex++) {
    imgdata[count] = scrren[index][subindex].red;
    count++;
    imgdata[count] = scrren[index][subindex].green;
    count++;
    imgdata[count] = scrren[index][subindex].blue;
    count++;
    imgdata[count] = scrren[index][subindex].contrast;
    count++;
  }
}
var screen_average = new Array(25);

console.log({ scrren });
console.log({ imgdata });
scrren[3][0].red = 1;
console.log(scrren[3][0].blue);
console.log({ imgdata });

function checkImageDate() {
  console.log(image_data.data);
}

function change_to_object() {
  var object = [];

  for (index = 0; index < image_data_copy.length; index += 4) {
    object.push({
      red: image_data_copy[index],
      green: image_data_copy[index + 1],
      blue: image_data_copy[index + 2],
      contrast: image_data_copy[index + 3],
    });
  }
  console.log({ object });
  orginal_pixels_copy = object.slice();
  return object;
}
```

```
var image_screen = new Array(source_image.width);
function change_to_screen() {
  for (index = 0; index < source_image.width; index++) {
    image_screen[index] = new Array(source_image.height);
    for (subindex = 0; subindex < source_image.height; subindex++) {
      image_screen[index][subindex] = orginal_pixels_copy[0];
      orginal_pixels_copy.shift();
    }
  }
  console.log(image_screen[0][0].red);
}

function get_average_image() {
  console.log({ image_screen });
  for (index = 1; index < source_image.width - 2; index++) {
    for (subindex = 1; subindex < source_image.height - 2; subindex++) {
      var red =
        (image_screen[index - 1][subindex - 1].red +
          image_screen[index][subindex - 1].red +
          image_screen[index + 1][subindex - 1].red +
          image_screen[index - 1][subindex].red +
          image_screen[index - 1][subindex + 1].red +
          image_screen[index][subindex + 1].red +
          image_screen[index + 1][subindex].red +
          image_screen[index + 1][subindex + 1].red) /
        8;
      var green =
        (image_screen[index - 1][subindex - 1].green +
          image_screen[index][subindex - 1].green +
          image_screen[index + 1][subindex - 1].green +
          image_screen[index - 1][subindex].green +
          image_screen[index - 1][subindex + 1].green +
          image_screen[index][subindex + 1].green +
          image_screen[index + 1][subindex].green +
          image_screen[index + 1][subindex + 1].green) /
        8;
      var blue =
        (image_screen[index - 1][subindex - 1].blue +
          image_screen[index][subindex - 1].blue +
          image_screen[index + 1][subindex - 1].blue +
          image_screen[index - 1][subindex].blue +
          image_screen[index - 1][subindex + 1].blue +
          image_screen[index][subindex + 1].blue +
          image_screen[index + 1][subindex].blue +
          image_screen[index + 1][subindex + 1].blue) /
        8;
      var contrast =
        (image_screen[index - 1][subindex - 1].contrast +
          image_screen[index][subindex - 1].contrast +
          image_screen[index + 1][subindex - 1].contrast +
          image_screen[index - 1][subindex].contrast +
          image_screen[index - 1][subindex + 1].contrast +
          image_screen[index][subindex + 1].contrast +
          image_screen[index + 1][subindex].contrast +
          image_screen[index + 1][subindex + 1].contrast) /
        8;

      image_screen[index][subindex] = {
        red: red,
        green: green,
        blue: blue,
        contrast: contrast,
      };
    }
  }
  console.log({ image_screen });
}

function get_median_image() {
  console.log({ image_screen });
  for (index = 1; index < source_image.width - 2; index++) {
    for (subindex = 1; subindex < source_image.height - 2; subindex++) {
      var red = new Array(7);

      red[0] = image_screen[index - 1][subindex - 1].red;
      red[1] = image_screen[index][subindex - 1].red;
      red[2] = image_screen[index + 1][subindex - 1].red;
      red[3] = image_screen[index - 1][subindex].red;
      red[4] = image_screen[index - 1][subindex + 1].red;
      red[5] = image_screen[index][subindex + 1].red;
      red[6] = image_screen[index + 1][subindex].red;
      red[7] = image_screen[index + 1][subindex + 1].red;
      var green = new Array(7);

      green[0] = image_screen[index - 1][subindex - 1].green;
      green[1] = image_screen[index][subindex - 1].green;
      green[2] = image_screen[index + 1][subindex - 1].green;
      green[3] = image_screen[index - 1][subindex].green;
      green[4] = image_screen[index - 1][subindex + 1].green;
      green[5] = image_screen[index][subindex + 1].green;
      green[6] = image_screen[index + 1][subindex].green;
      green[7] = image_screen[index + 1][subindex + 1].green;
      var blue = new Array(7);

      blue[0] = image_screen[index - 1][subindex - 1].blue;
      blue[1] = image_screen[index][subindex - 1].blue;
      blue[2] = image_screen[index + 1][subindex - 1].blue;
      blue[3] = image_screen[index - 1][subindex].blue;
      blue[4] = image_screen[index - 1][subindex + 1].blue;
      blue[5] = image_screen[index][subindex + 1].blue;
      blue[6] = image_screen[index + 1][subindex].blue;
      blue[7] = image_screen[index + 1][subindex + 1].blue;
      var green = new Array(7);

      contrast[0] = image_screen[index - 1][subindex - 1].contrast;
      contrast[1] = image_screen[index][subindex - 1].contrast;
      contrast[2] = image_screen[index + 1][subindex - 1].contrast;
      contrast[3] = image_screen[index - 1][subindex].contrast;
      contrast[4] = image_screen[index - 1][subindex + 1].contrast;
      contrast[5] = image_screen[index][subindex + 1].contrast;
      contrast[6] = image_screen[index + 1][subindex].contrast;
      contrast[7] = image_screen[index + 1][subindex + 1].contrast;

      image_screen[index][subindex] = {
        red: red[1],
        green: green[1],
        blue: blue[1],
        contrast: contrast[1],
      };
    }
  }
  console.log({ image_screen });
}

function get_new_image_data() {
  bob = new Uint8ClampedArray(image_data.data.length);

  var count = 0;
  for (index = 0; index < source_image.width; index++) {
    for (subindex = 0; subindex < source_image.height; subindex++) {
      bob[count] = image_screen[index][subindex].red;
      count++;
      bob[count] = image_screen[index][subindex].green;
      count++;
      bob[count] = image_screen[index][subindex].blue;
      count++;
      bob[count] = image_screen[index][subindex].contrast;
      count++;
    }
  }
  commit_2();
  return bob;
}

function timer() {
  setInterval(function () {
    get_average_image();
    get_new_image_data();
    commit_2();
  }),
    1;
}
```