**"Absence and Presence"**

**A Printmaking Project for Al-Mutanabbi Street**

In 2013, printmakers from around the world were invited to make prints for the Al-Mutanabbi Street Project.  Each artist was asked to reflect and respond to the bombing of the bookseller’s quarter in Baghdad in 2007 and to be part of a global coalition of artists, poets and writers that states that wherever people talk freely and creativity breathes, Al Mutanabbi Street starts.

Printmaking is a physical process that leaves an imprint and yet retains its ability to recede, impress, layer and renew. In this vein, each printmaker will donate five prints from their edition to the project. As in our broadside and artist book project, one complete set of prints will be donated to the Iraq National Library in Baghdad. Two copies will join exhibits in the U.S. One copy will go to the UK (and be used as part of exhibits there and Europe, the Middle East and North Africa). One copy (in 2016) will join work from the project at the Herron Art Library (Indiana - Purdue University) and be digitized by them, becoming part of their permanent collection of work from this project. Dialog related to the exhibitions of the prints from "Absence and Presence" will begin in December of 2014 at the inaugural exhibition at the San Francisco Center for the Book.

Additional exhibitions will be held in the US, UK, Middle East and North Africa from 2015-2017.

<http://www.al-mutanabbistreetstartshere-boston.com/printmaking-project.html>

**Participating Artist came from these Countries**

Australia

Argentina

Belgium

Canada

Denmark

France

Germany

India

Iran

Ireland

Iraq

Italy

Lithuania

Malta

Netherlands

Peru

Poland

Romania

South Korea

Spain

Switzerland

Sweden

Thailand

Turkey

United Kingdom

United States

Artist: **Andy Gossett**

City: **London UK**

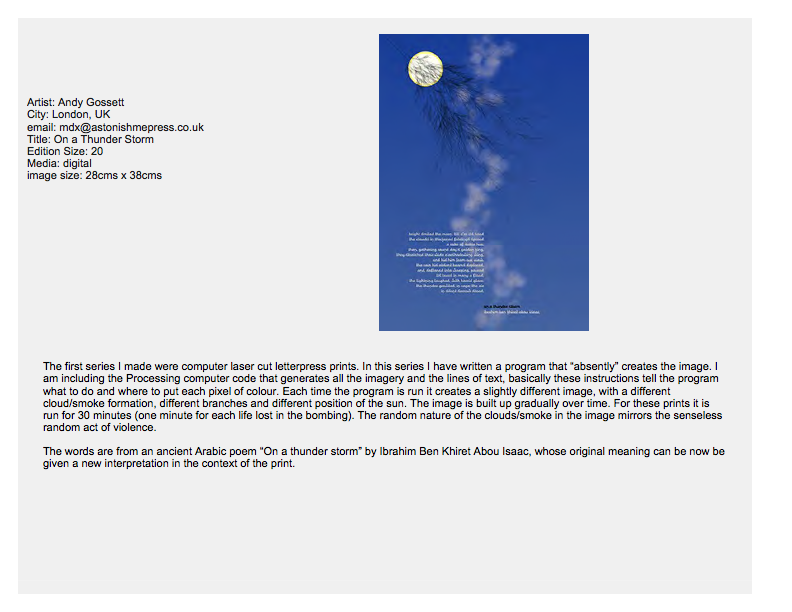
email: [**mdx@astonishmepress.co.uk**](mailto:mdx@astonishmepress.co.uk)

Title: **On a thunder Storm**

Edition Size: **20**

Media: **Digital**

Image size: **28cm x 38cm**

The first series I made were computer laser cut letterpress prints. In this series I have wriiten a program that “absently” creates the image. I am including the Proceesing computer code that generates all the imagery and the lines of text, basically these instructions tell the program what to do and where to put each pixel of colour. Each time the program is run it creates a slightly different image, with a different cloud/smoke formation, different branches and different position of the sun. The image is built up gradually over time. For these prints it is run for 30 minutes (one minute for each life lost in the bombing). The random nature of the clouds/smoke in the image mirrors the senseless random act of violence.

The words are from an ancient Arabic poem “On a thunder storm” by Ibrahim Ben Khiret Abou Isaac, whose original meaning can be now be given a new interpretation in the context of the print.

*The following pages show examples of the images created each time the code is run and the code itself.*

*Page from the entire artist collection which can be downloaded here:*

https://hubic.com/home/pub/?ruid=aHR0cHM6Ly9sYjEwNDAuaHViaWMub3ZoLm5ldC92MS9BVVRIXzU2MGJkM2FjN2MwYmYxY2Q5Mjc5MDIxMGVhOGE5NmQ2L2RlZmF1bHQvLm92aFB1Yi8xNDM0MDQwOTQwXzE0MzY2MzI5NDA/dGVtcF91cmxfc2lnPWZiZjJmYmQ0MjNkYzBlODVkMzczNDg2ZDRmYTkyZjc2ZGFmNGU5NmYmdGVtcF91cmxfZXhwaXJlcz0xNDM2NjMyOTQw#

**Processing code**

// The MIT License (MIT)

// Copyright (c) 2014 Andy Gossett

// on\_a\_thunder\_storm

int Y\_AXIS = 1;

// this is used later to create the graduation in the background colour

// using the lerp function

color c1, c2;

int bw = 10;

PFont f;

color[] palette = {

#FCFEFF, #EBF0F7, #E8F2FF, #FAFCFF, #E8EFF7

};

color[] palette1 = {

#E1E1E8, #ADABBF, #ADABBF, #4F4D67, #BBB6EA

};

float ra = random(50);

int newR = int(ra)\*10;

int[] start = {

10, 30, 50, 70, 90, 110, 130, 190, 200, 250, 300, 400, 550

};

// this is used later to create the pause function

boolean bStop;

class cloud\_rambler {

int x, y;

cloud\_rambler() {

x = width/2;

// y = height/2;

// y = height/4;

y = height/5;

}

void render() {

noStroke();

float r = random(5);

fill(palette1[int(r)], 10);

ellipse(x, y, r\*40, r\*40);

}

void step() {

// f = loadFont(“Talib-KulkufiRounded-96.vlw”);

f = loadFont(“Talib-MohandesRegular-200.vlw”);

textFont(f);

// we select the colour of the font

fill(200, 200, 200);

textAlign(LEFT);

// here we add the text, this must be inside double quotes

// followed by the X and Y position.

textSize(60);

fill(0);

text(“On a thunder storm”, width/2, height-390);

fill(200, 200, 200);

text(“Ibrahim Ben Khiret Abou Isaac”, width/2, height-300);

textSize(60);

textAlign(RIGHT);

text(“Bright smiled the morn, till o’er its head\nThe clouds in thickened foldings spread\nA robe of sable hue;\nThen, gathering round Day’s golden King,\nThey stretched their wide o’ershadowing wing,\nAnd hid him from our view.\nThe rain his absent beams deplored,

\nAnd, softened into weeping, poured

\nIts tears in many a flood;\n

The lightning laughed, with horrid glare;\nThe thunder growled, in rage; the air\nIn silent sorrow stood.”, width/2, height-1600);

// this sets up the random wandering of the cloud ellipses

int choice = int(random(4));

if (choice == 0) {

x = x+40;

} else if (choice == 1) {

x = x-40;

} else if (choice == 2) {

y= y+40;

} else {

y=y-40;

}

x = constrain(x, 0, width-80);

y = constrain(y, 0, height-80);

}

}

cloud\_rambler c;

void setup() {

// this size is really big, too big to fit onto a laptop sceen

// but is needed to generate the high resolution A3 300 dpi image

size(3500, 4960);

// this defines the start and end colours in the background range

c1 = color(25, 60, 150);

c2 = color(70, 100, 160);

// this sets up the gradient and the direction (vertically, so Y\_AXIS)

setGradient(0, 0, width, height, c1, c2, Y\_AXIS);

}

void setGradient(int x, int y, float w, float h, color c1, color c2, int axis ) {

noFill();

if (axis == Y\_AXIS) { // Top to bottom gradient

for (int i = y; i <= y+h; i++) {

float inter = map(i, y, y+h, 0, 1);

color c = lerpColor(c1, c2, inter);

stroke(c);

line(x, i, x+w, i);

}

}

// make a sun, from 6 ellipses, gettin smaller and brighter

float sun = random(width/8, width/2);

noStroke();

fill(250, 230, 100, 255);

ellipse(sun, width/6, width/6, width/6);

noStroke();

fill(250, 240, 130, 255);

ellipse(sun, width/6, width/6-(20), width/6-(20));

noStroke();

fill(250, 245, 150, 255);

ellipse(sun, width/6, width/6-(40), width/6-(40));

noStroke();

fill(255, 255, 200, 255);

ellipse(sun, width/6, width/6-(60), width/6-(60));

noStroke();

fill(255, 255, 240, 255);

ellipse(sun, width/6, width/6-(80), width/6-(80));

// this defines the colour and stroke weight of the branches

fill(0, 0, 255);

strokeWeight(4);

stroke(0, 0, 0, 60);

smooth();

noStroke();

c = new cloud\_rambler();

}

void draw() {

// run cloud\_rambler object

c.step();

c.render();

float r = random(10);

Vector seed = new Vector(start[int(r)], 0, 200, 45);

fractal (seed, DEPTH );

}

void keyPressed() {

if (key == ‘ ‘) {

bStop = !bStop;

if (bStop)

noLoop();

else

loop();

}

if (key == ‘x’ || key == ‘X’) {

background(25, 100, 200);

// make the sun

float sun = random(width/8, width/2);

noStroke();

fill(250, 230, 100, 255);

ellipse(sun, width/6, width/6, width/6);

noStroke();

fill(250, 240, 130, 255);

ellipse(sun, width/6, width/6-(20), width/6-(20));

// ellipse(sun, 100, 135, 135);

noStroke();

fill(250, 245, 150, 255);

ellipse(sun, width/6, width/6-(40), width/6-(40));

// ellipse(sun, 100, 130, 130);

noStroke();

fill(255, 255, 200, 255);

ellipse(sun, width/6, width/6-(60), width/6-(60));

// ellipse(sun, 100, 125, 125);

noStroke();

fill(255, 255, 240, 255);

ellipse(sun, width/6, width/6-(80), width/6-(80));

// ellipse(sun, 100, 90, 90);

}

if (key == ‘s’ || key == ‘S’) {

saveFrame();

}

if (key == ‘b’ || key == ‘B’) {

fill(0, 0, 255);

stroke(0, 0, 0, 60 );

smooth();

float r = random(10);

Vector seed = new Vector(start[int(r)], 0, 200, 45);

// Vector seed = new Vector(start[int(r)], 0, 200, random(80));

fractal (seed, DEPTH );

}

}

int BRANCHES = 4; //Number of branches per line

int DEPTH = 5; // Recursive depth

float MIN\_ANGLE = 10.0; //Minimum angle for new branch

float MAX\_ANGLE = 25.0; //Maximum angle for new branch

//float MAX\_ANGLE = 15.0; //Maximum angle for new branch

float MIN\_LENGTH = 0.30; //Minimum length of new branch, as a pct of current length

float MAX\_LENGTH = 0.85; //Maximum length of new branch, as a pct of current length

// Implements a Vector

class Vector {

int x, y;

float r, theta;

Vector (int \_x, int \_y, float \_r, float \_theta) {

x = \_x; //Origin x

y = \_y; //Origin y

// r = \_r\*1.5; //Length

r = \_r\*7; //Length

theta = \_theta; // Angle

}

int getEndPointX() {

return int(x + r\*cos(theta/57.3));

}

int getEndPointY() {

return int(y + r\*sin(theta/57.3));

}

}

//Recursive function that creates a fractal “plant”

void fractal(Vector v, int N) {

if (N > 0) {

int dir = 1; //control alternating direction of the branch

line(v.x, v.y, v.getEndPointX(), v.getEndPointY()); //Draw the current vector

for (int i = 0; i < BRANCHES; i++) {

//Create a random vector based on the current one

Vector r = new Vector (v.x, v.y, v.r, v.theta); //New random vector that will branch off the current line

r.r = random(v.r\*MIN\_LENGTH, v.r\*MAX\_LENGTH); //Select a random length

r.x = r.getEndPointX(); //shift the x-origin

r.y = r.getEndPointY(); //shift the y-origin

r.theta += dir\*random(MIN\_ANGLE, MAX\_ANGLE); // shift the angle a bit

dir = dir \* -1; //Alternate branch direction

fractal(r, N-1); //Recurse

}

}

}



**An Inventory of Al-Mutanabbi Street Exhibits 2012-2016: Artists' Books and Prints**

**List of Exhibitions**

**2016**

George Mason University, Fairfax, Virginia

January - March, 2016\*\*\*\*

(This is a Washington,D.C. city wide effort involving other exhibit spaces, organizations, and other universities as well)

Curry College Gallery, Milton, Massachusetts\*

August 28-October 23 (prints 1-25)

November 2-January 2 (prints 26-50)

 First reception: September 10, 2015

 Second reception: November 5, 2015

The Exeter Central Library, Exeter (UK)

March 2016\* (Prints)

The Multnomah County Library, Portland, Oregon

March 5, - May 15, 2016 \*\*\*

The San Francisco Public Library Gallery

September 15 – December 15, 2016. \*

City College Library, San Francisco, California

October 24, 2016 through April 16, 2017.\*

The Frank & Katrina Basile Gallery at the Herron School of Art and Design-

The Herron Art Library of IUPUI University

November 2016\*\*\*\*

**2015**

Arab American National Museum- Dearborn/Detroit, Michigan  March 6- July 12, 2015 \*\*

Hampshire College, Amherst, Massachusetts

-June 8 - September 30, 2015\*\*

Idaho Center for the Book in partnership with The Arts and Humanities Institute at Boise State University

-October 1, 2015- January 30,  2016 \*

Curry College Gallery, Milton, Massachusetts

November - January (2015-2016: 25 prints)

\* designates exhibits of fifty or less (often in combinations) of letterpress broadsides, artists' books , and/or prints  
\*\* designates exhibits of between fifty and one hundred and fifty (often in combinations) of artists' books, broadsides, and/or prints

\*\*\* designates exhibits of between one hundred and fifty and two hundred (often in combinations) of letterpress broadsides, artists' books, and/or prints

\*\*\*\* designates exhibits of the complete project collection of letterpress broadsides, artists' books, and prints.

http://www.al-mutanabbistreetstartshere-boston.com/united-kingdom.html

http://www.al-mutanabbistreetstartshere-boston.com/exhibitions.html

**Articles**

**Washington Post**

<http://www.washingtonpost.com/lifestyle/style/author-azar-nafisi-understand-countries-in-turmoil-by-learning-their-history/2011/07/11/gIQA4NIWGI_story.html>

**Foreign Policy In Focus**

<http://fpif.org/recreating_baghdads_lost_literary_street/>

<http://www.al-mutanabbistreetstartshere-boston.com/>

<http://www.al-mutanabbistreetstartshere-boston.com/remembering-al-mutanabbi-by-thomas-christensen.html>

<http://www.al-mutanabbistreetstartshere-boston.com/al-mutanabbi-street-by-lutfia-alduleimi.html>