

MINISTERUL EDUCAȚIEI ȘI CERCETĂRII AL REPUBLICII MOLDOVA
Universitatea Tehnică a Moldovei
Facultatea Calculatoare, Informatică și Microelectronică
Departmentul Inginerie Software și Automatică

Admis la susținere
Șef departament: Fiodorov I. dr., conf.univ.

„___” _____ 2022

**Analiza algoritmilor și implementarea artei firului
de ață în cadrul aplicației web**

Teză de master

Student: _____ **Lîniuc Cristina, gr. IS-211M**

Coordonator: _____ **Gavrilița Mihail, asist.univ**

Consultant: _____ **Catruc Mariana, lect. univ.**

Chișinău, 2023

Rezumat

În zilele noastre, sectorul divertismentului joacă un rol imens în economie și societate, unde oamenii își petrec timpul jucându-se, învățând noi abilități într-un mod ludic. Pe de altă parte, timpul liber este plin de lucrări lucrate manual. În ultimii ani, trusele gata de fabricat au devenit din ce în ce mai populare, cum ar fi pictura pe numere, pictura cu diamant. Datorită tehnologiilor în evoluție, lucrarea manuală a devenit mai personalizabilă precum pictarea portretului persoanei iubite. Această lucrare abordează o nouă meșteșug, arta șirurilor, care le permite utilizatorilor finali să-și personalizeze munca în modul în care doresc.

Această lucrare descrie dezvoltarea aplicației web de creație artizanală concepută pentru generarea algoritmului de realizare a artei șirurilor. Unul dintre obiectivele lucrării este de a implementa cel mai bun algoritm cu cel mai precis rezultat. Scopul principal al algoritmului implementat este de a găsi intrările de date adecvate care vor oferi cea mai bună imagine rezultată.

Lucrarea prezintă, de asemenea, procesul de dezvoltare a unei aplicații web care permite utilizatorilor să o acceseze de pe fiecare dispozitiv și locație. Aplicația va permite utilizatorului să încarce imaginea sau fotografia dorită, să aleagă intrările dorite, cum ar fi numărul de pini, dimensiunea pânzei și rezoluția imaginii de ieșire. Ca rezultat, utilizatorul va primi instrucțiunile pentru asamblarea imaginii.

Abstract

Nowadays, the entertainment sector plays a huge role in the economy and society, where people spend their time playing games, learning new skills in a playful way. On the other hand, leisure time is filled with handmade work. In the last years, ready-to-make kits become more and more popular, like painting on numbers, diamond painting. Due to the evolving technologies, handcrafting became more customizable like painting the portrait of the loved person. This paper tackles a new craft, string art, which allow end-users to customize their work in the way they want

This paper describes the development of craft-making web application designed for generating the algorithm for string art making. One of the goals of the paper is to implement the best algorithm with the most accurate result. The main purpose of the implemented algorithm is to find the appropriate data inputs that will give the best resulting image.

The paper also presents the development process of a web application that allow its' users to access it from every device and location. The application will permit the user to upload the wanted image or photo, to choose the desired inputs like the number of pins, the size of the canvas and the resolution of the output image. As a result, the user will get the instructions for assembling the picture.

Table of contents

List of figures	8
Listings	9
INTRODUCTION	10
1 DOMAIN ANALYSIS	11
1.1 Motivation	11
1.2 Solutions on the market	14
1.3 Objectives	16
1.4 Desired functionalities	18
2 ALGORITHM ANALYSIS	20
2.1 Generative art	20
2.2 Algorithm analysis	24
2.2.1 Digital differential analyzer algorithm	25
2.2.2 Bresenham line algorithm	27
2.2.3 The aliasing problem	29
2.3 Algorithm implementation	31
2.4 Results and Similarity comparison	34
3 APPLICATION DEVELOPMENT	40
3.1 Used technologies	40
3.2 System development architecture	43
3.3 Software development	49
Conclusions	55
References	56

List of Figures

1.1	String art generation process [4]	17
2.1	Three Ways to render, “Draw a horizontal line across the center of the canvas.” [3]	21
2.2	A 2D line as a path where a) y-axis is pointing upwards b) y-axis is pointing downwards.” [11]	25
2.3	Criterion on how to select a pixel	26
2.4	Use distances to determine which pixel to color in column $x_k + 1$ [15]	29
2.5	String art result	34
2.6	Results of different strings number: 1500, 2500, 3000, 4000, 6000	34
2.7	String art on photo: original, 2000, 4000, 6000	35
2.8	String art on white and black canvas	35
2.9	String art:Original, black&white, color	35
2.10	Mean Squared Error	36
2.11	Structural Similarity Index	37
2.12	Mean Squared Error and Structural Similarity Index on image output	38
2.13	Face similarity comparison	38
2.14	Similarity comparison on 6000 strings	39
3.1	4+1 architecture [24]	43
3.2	Deployment diagram	44
3.3	Component diagram	46
3.4	Registration to the service	47
3.5	Registration to the service	48
3.6	Class diagram	48
3.7	Registration to the service	49
3.8	Main page	53
3.9	Main page - instruction generation	53
3.10	Registration page	54

Listings

2.1	Transform coloured image to grayscale	31
2.2	Transform coloured image to grayscale	32
2.3	Nail positions on canvas	32
2.4	Anti-aliased coordinates function	32
2.5	Best nail positions	33
2.6	Image creation function	33
2.7	Mean Squared Error calculation method	36
2.8	Structural Similarity Index method	37
2.9	Structural Similarity Index method	37
3.1	User model	50
3.2	Image model	50
3.3	Image table schema	51
3.4	Image controller	51
3.5	Method for image upload	52
3.6	Method for instructions generation	52

INTRODUCTION

String art is a technique for the creation of visual artwork where images emerge from a set of strings that are spanned between pins. There are several artists that have experimented with the creation of string art using a proprietary algorithmic solutions in combination with manual crafting or automatic fabrication. In this paper, it is proposed a formal treatment of the problem and an automatic computational fabrication pipeline to generate string images that resemble a gray scale image given as input.

The goal of the work is to compute the string path around the given pins and to fabricate the string images. While the pins can be placed in any pattern, the focus is on a setup where pins are distributed along a circular frame which encloses a canvas. In addition, the image will be generated by a single string without cutting it.

The main challenge of computational string art is the proper formulation of an optimization problem that computes the path of the string. There are many parameters and design choices leading to very different optimization problems. For example, the selection of parameters, like the number of pins and the image resolution used in the computation, drive the scale of the problem. The image formation model influences the complexity of optimization. Considering all possible factors leads to an overly ambitious modeling of the problem that is discrete, large scale, and non-linear. Therefore, the best known algorithm for generating string art proposes to relax the problem into a least squares formulation and employ thresholding. This leads to a reasonable baseline solution, but there are still too many visual artifacts.

References

- 1 SACK, Warren. *The Software Arts (Software Studies)*. The MIT Press; First Edition (April 9, 2019). ISBN 0262039702
- 2 WILSON, Stephen. *Information Arts: Intersections of Art, Science, and Technology*. The MIT Press; Reprint edition (February 28, 2003). ISBN 0262731584
- 3 PARKER, James R., Author; DIAMOND, Sara L., Foreword *Generative art: Algorithms as artistic tool*. UpRoute (December 15, 2019). ISBN 978-1-9888243-8-3
- 4 BIRSAK, Michael; RIST, Florian; WONKA, Peter; MUSIALSKI, Przemyslaw; *String Art: Towards Computational Fabrication of String Image*. EUROGRAPHICS 2018 / D. Gutierrez and A. Sheffer. Disposable: https://www.researchgate.net/publication/322766118_String_Art_Towards_Computational_Fabrication_of_String_Images [cited 20.09.2022]
- 5 VRELLIS, Petros. *Experimental / computational string art*. [cited 26.09.2022] Disposable: <http://artof01.com/vrellis/works/knit.html>
- 6 *RingString solution*, ringstring.shop. [cited 26.09.2022] Disposable: <https://ringstring.shop/>
- 7 *ArtRapid solution*, artrapid.com ©2022. [cited 26.09.2022] Disposable: <https://artrapid.com/>
- 8 *Stringading solution*, stringading.com ©2022. [cited 26.09.2022] Disposable: <https://www.stringading.com/blog/online-string-art-generator>
<https://www.stringading.com/blog/online-string-art-generator>
- 9 *Thead Art (String Art) solution*, Google Play. [cited 28.09.2022] Disposable: <https://play.google.com/store/apps/details?id=in.fourT.stringart>
- 10 *Alinedeco solution*, make.alinedeco.com. [cited 30.09.2022] Disposable: <http://make.alinedeco.com/design/>
- 11 ELIAS, Rimón; *Digital Media: A Problem-solving Approach for Computer Graphics*. Springer International Publishing Switzerland 2014. ISBN 978-3-319-05139-9
- 12 AGOSTON, Max K.; *Computer Graphics and Geometric Modeling: Implementation and Algorithm*. Springer-Verlag London Limited 2005. ISBN 1-85233-818-0

- 13 *DDA Line Drawing Algorithm :Explanation of the DDA Line Drawing Algorithm with examples*, medium.com. Disposable:<https://medium.com/geekculture/dda-line-drawing-algorithm-be9f069921cf> [cited 5.10.2022]
- 14 Zongli Shi. *Graphics Programming. Principles and Algorithms*. Disposable:<https://www.whitman.edu/Documents/Academics/Mathematics/2017/Shi.pdf> [cited 20.10.2022]
- 15 *Tutorialspoint.com. Line Generation Algorithm*. Disposable: https://www.tutorialspoint.com/computer_graphics/line_generation_algorithm.htm. [cited 22.10.2022]
- 16 *Patterns causing your image to glitch? That's a moiré problem.*, Adobe. Disposable:<https://www.adobe.com/creativecloud/photography/discover/anti-aliasing.html> [cited 27.10.2022]
- 17 Allan V. Oppenheim. *Discrete-Time Signal Processing*. Prentice Hall, 2 edition, 1990. ISBN 0137549202.
- 18 Franklin C. Crow. *The aliasing problem in computer generated shaded images*. volume 20. Communications of the ACM, November 1977
- 19 KESTEN, Victor. *Evaluating Different Spatial Anti Aliasing Techniques* EXAMENSARBETE TEKNIK, GRUNDNIVÅ, 15 HP, STOCKHOLM SVERIGE 2017. Disposable:<https://kth.diva-portal.org/smash/get/diva2:1106244/FULLTEXT01.pdf> [cited 15.11.2022]
- 20 *Ruby on Rails*. Wikipedia. [cited 15.11.2022] Disposable: https://en.wikipedia.org/wiki/Ruby_on_Rails
- 21 *Devise Ruby Gem*. Github. [cited 28.11. 2022] Disposable: <https://github.com/heartcombo/devise>
- 22 *Sidekiq Ruby Gem*. sidekiq.org. [cited 29.11.2022] Disposable: <https://sidekiq.org/>
- 23 *Sidekiq*. Wikipedia. [cited 29.11.2022] Disposable: <https://en.wikipedia.org/wiki/Sidekiq>
- 24 KRUCHTEN, Philippe. *Architectural Blueprints — The “4+1” View Model of Software Architecture*. In: *IEEE Software 12 (6), pp. 42-50*. [cited 15.02.2021] Disposable: <https://www.cs.ubc.ca/~gregor/teaching/papers/4+1view-architecture.pdf> [cited 5.12.2022]
- 25 MILES, Russ, HAMILTON Kim. *Learning UML 2.0*. O'Reilly Media; 1st edition, 2006. ISBN 0596009828