Generative type design: An approach focused on skeletons extraction and their anatomical deconstruction*

Jéssica Parente
CISUC, UNIVERSITY OF COIMBRA
PORTUGAL

Tiago Martins
CISUC, UNIVERSITY OF COIMBRA
PORTUGAL

João Bicker
CISUC, UNIVERSITY OF COIMBRA
PORTUGAL

THEMATIC AREAS
TYPE DESIGN; TYPOGRAPHY AND MULTIMEDIA DESIGN

KEYWORDS
TYPE DESIGN; TYPOGRAPHY; GENERATIVE DESIGN; COMPUTATIONAL DESIGN

▶ Typography is a way of visualizing language (Cheng, 2005). For designers, typography has an even more prominent role since it also provides a layer of content. The choice and use of a font gives clues regarding the subject being addressed. In order to use typography in the best way, many designers study their anatomy and ways to categorize it.

With the digital revolution appeared the personal computers and tools that came to facilitate the design of fonts. As a result, more fonts appeared, but with them the uncertainty of their quality.

It is necessary to create tools that established a balance between what the user can determine and what the system does autonomously. However, the society in which we live is dependent on personalization. Suggestions according to our views become increasingly a requirement on social networks.

In order to answer these questions we came up with the idea of a computer system that would allows designing fonts in a generative way. In the first phase an investigation was made on (i) typographic anatomy and classification and (ii) a review of the state of the art in relation to the dynamic and generative drawing of typefaces. Subsequently, the computer system was developed. For this, three aspects were worked out: (i) he development of the generated typefaces' structure and the codification of the different elements of the letter's anatomy in multiple layers; (ii) the combination of layers of different typefaces; and (iii) the creation of fonts through the generation/ modification these layers' elements. This approach will allow the generation of glyphs automatically and coherently.

* This article results from the work developed for the dissertation "Desenho Generativo de Tipos de Letra"; Jéssica Parente; 2018

▶ A tipografia é uma forma de visualizar a linguagem (Cheng, 2005). Para os designers, ganha ainda mais importância, pois acrescenta uma camada de conteúdo. A escolha e utilização de um tipo de letra dá indicações relativas ao assunto que estamos a abordar. Com o intuito de utilizar a tipografia da melhor forma, muitos designers estudam a sua anatomia e formas de a categorizar.

Com a revolução digital surgiram os computadores pessoais e as ferramentas que vieram facilitar o desenho de tipos de letra. Consequentemente, surgiram mais tipos de letra, mas com eles a incerteza da sua qualidade.

É necessário criar ferramentas que estabeleçam um equilíbrio entre o que o utilizador pode determinar e o que o sistema faz autonomamente. Por outro lado, a sociedade em que vivemos está dependente da personalização. Sugestões de acordo com as nossas visualizações tornam-se cada vez mais um requisito nas redes sociais.

Com o objetivo de responder a essas questões surgiu-nos a ideia de um sistema computacional que permitisse o desenho de tipos de letra de forma generativa. Para isto foram trabalhadas três vertentes: (i) o desenvolvimento da estrutura dos tipos de letra gerados e codificação dos diferentes elementos da anatomia da letra em diferentes camadas; (ii) a combinação de camadas de diferentes tipos de letra; e (iii) a criação de tipos de letra através da geração/modificação dos elementos destas camadas. Esta abordagem permitirá a geração de glifos de forma automática e coerente.

Desenho generativo de tipos de letra: Uma abordagem focada na extração de esqueletos e desconstrução anatómica*

Jéssica Parente
CISUC, DEPARTAMENTO DE ENGENHARIA INFORMÁTICA, UNIVERSIDADE DE COIMBRA, 3030 COIMBRA, PORTUGAL

Tiago Martins
CISUC, DEPARTAMENTO DE ENGENHARIA INFORMÁTICA, UNIVERSIDADE DE COIMBRA, 3030 COIMBRA, PORTUGAL

João Bicker
CISUC, DEPARTAMENTO DE ENGENHARIA INFORMÁTICA, UNIVERSIDADE DE COIMBRA, 3030 COIMBRA, PORTUGAL

ÁREA CIENTÍFICA
DESIGN DE TIPOS; TIPOGRAFIA E DESIGN MULTIMÉDIA

PALAVRAS-CHAVE
DESENHO DE TIPOS, TIPOGRAFIA, DESIGN GENERATIVO, DESIGN COMPUTACIONAL

* Este artigo resulta do trabalho desenvolvido para a dissertação "Desenho Generativo de Tipos de Letra"; Jéssica Parente; 2018

1. Introdução

A forma como comunicamos é uma das características que mais nos definem como seres humanos. No nosso dia-a-dia utilizamos as duas vertentes da linguagem, escrita ou falada, por necessidade ou conveniência.

> *Type is the visual manifestation of language. It is instrumental in turning characters into words, and words into messages.*
> Karen Cheng, Designing Type (2005)

Como Cheng refere, a tipografia é uma forma de visualizar a linguagem. Adrian Schaughnessy acrescenta que esta é também uma forma de dar "significado". Já não é suficiente que a tipografia seja tipografia. Tem que transmitir ressonância e profundidade às mensagens que está a transmitir. O seu desenho pode, portanto, ser usado de forma a acrescentar camadas de conhecimento (Cheng, 2005; Schaughnessy, 2009).

Com o avanço tecnológico as ferramentas de tipografia sofreram alterações. A tecnologia veio possibilitar novas formas de exploração e permitir ao designer de tipos a exploração de campos anteriormente impensáveis. Consequentemente, através destas novas possibilidades, surgem tipos de letra que se adaptam a diversos contextos (Lehni, 2011).

Com o intuito de tirar partido dessas novas possibilidades surgiu-nos a ideia de um sistema computacional que permitisse o desenho de tipos de letra de forma generativa. Pretendiamos que os tipos de letra gerados fossem criados através da conjugação de diversas camadas — elementos da anatomia da letra — e a sua posterior modificação. Depois de estabelecidas as camadas, o objetivo era a sua combinação e a criação de um método para o seu preenchimento. Além disso pretendíamos que glifos gerados acrescentassem uma camada de conhecimento e, portanto, era essencial que o preenchimento das camadas não fosse fixo. O sistema deveria permitir a alteração de parâmetros e, desse modo, cada tipo de letra gerado poderia ser alterado se aplicado numa certa circunstância. Pretendíamos criar um equilíbrio entre a tipografia tradicional, a tipografia como narrativa e a generatividade para o design e, neste caso mais específico, para os tipos de letra.

2. Trabalho relacionado

Inicialmente, a tipografia era vista como um sistema constituído por uma série de regras. No entanto, a chegada dos movimentos artísticos do início do século XX estabelece uma visão diferente. Os artistas de vanguarda rejeitavam as formas históricas e olhavam para o alfabeto como um sistema de relações abstratas. Esse aspecto veio a intensificar-se com o surgimento do Futurismo e do Dadaísmo. As composições tipográficas criadas pretendiam criar uma reação emocional no leitor. Mais tarde, com a chegada dos anos 80 assistimos a uma revolução direta contra as ideias de legibilidade e racionalidade. Uma nova linguagem visual foi criada a partir da desconstrução do modernismo e surgiram designers e artistas como David Carson e Ed Fella (Lupton, 2006).

Com o aparecimento das novas tecnologias e da programação aplicada ao design surgiram fontes que marcaram uma nova era no design de tipos. O tipo de letra *Beowulf*, desenvolvido por Erik von

Blokland e Just van Rossum, apareceu juntamente com a primeira série de fontes com contornos aleatórios e comportamento programado. Uma das características mais peculiares era o facto de não existirem dois glifos idênticos, devido ao movimento aleatório de pontos dos contornos (Lupton, 2006; FontFont, n.d.a).

Em 2004, foi desenvolvido o *GenoTyp*, um sistema que gera fontes através da combinação de características genéticas de diferentes tipos de letra. As características das letras analisadas são codificadas através de *software* em sistemas hereditários. O sistema permite a combinação de fontes diferentes e manipulação dos seus genomas.

Em 2008, foi desenvolvida a *Bastard*, outra aplicação que cria tipos generativos da autoria de Tobias Tshense. Os tipos são gerados através da combinação de fragmentos de fontes muito diferentes e, portanto, cada letra gerada é sempre diferente dos resultados anteriores (Mainz, n.d.a).

No mesmo contexto, Stefanie Oppenhäuser desenvolveu, também em 2008, um sistema que permitia a combinação de três fontes distintas denominado *Fontmixer*. A aplicação permite ao utilizador a seleção dos tipos de letra e determinar se as fontes são adicionadas ou subtraídas umas às outras. A combinação das fontes é sempre diferente e aleatória (Mainz, n.d.b).

FF ThreeSix é um sistema experimental de tipos de letra ópticos, composto por seis tipos, em oito pesos desenvolvido pela dupla MuirMcNeil. O sistema ThreeSix opera em cinco funções tipográficas: (i) contorno, a forma das letras individuais; (ii) modulação do traço, ou seja, o equilíbrio óptico entre traços horizontais e verticais; (iii) junções, os efeitos ópticos nas interseções dos traços; (iv) peso, a acomulação de densidade nos esqueletos das formas das letras e (v) espaçamento, o ajuste das formas em sequências (FontFont, n.d.b; MuirMcNeil, n.d.a).

Intersect é outro sistema tipográfico desenvolvido por MuirMcNeil. Este sistema afasta-se da criação tipográfica tradicional no sentido em que o contraste deixa de ser binário (preto e branco; forma e contraforma). Os tipos de letra Intersect são constituídos por uma gama de pesos que cria a ilusão de densidade (MuirMcNeil, n.d.b).

Em 2011, a designer gráfica Dina Silanteva, iniciou um projeto de investigação em tipografia generativa denominado *Typographic Music* que foi aplicado à teoria musical. O projeto levou à criação de um sistema de identidade para um festival de música. Utilizava uma grelha básica com três formas geométricas simples para o desenho das letras. De acordo com as regras aplicadas, eram modificados uma série de parâmetros (raio, saturação, cor, transparência) (Silanteva, n.d.; Sullivan, 2011).

3. Experimentação

Este capítulo apresenta os métodos e experiências usados no contexto deste projeto. Está dividido em três seções que marcam as principais etapas de desenvolvimento. A primeira é referente à criação de uma estrutura base para os glifos gerados. A segunda etapa refere-se ao cruzamento de esqueletos, ou seja, à combinação de diferentes partes anatómicas constituintes dos diferentes esqueletos extraídos. Na última seção é abordado o método que utilizamos para dar corpo aos esqueletos finais.

3.1. Extração do esqueleto

À partida já estava estabelecido como seria a geração dos tipos de letra. No entanto, existiam ainda muitos aspectos desconhecidos. Queríamos combinar diversas partes anatómicas, mas não tínhamos um esqueleto onde as pudéssemos associar. Além disso, ainda não sabíamos de que forma iríamos separar as diferentes camadas. Portanto a primeira preocupação foi a estrutura básica das fontes geradas.

Antes de mais decidimos estabelecer os pontos indispensáveis para o esqueleto gerado. Desse modo, poderíamos validá-lo facilmente. Identificamos como pontos essenciais para este esqueleto: (i) identificação do glifo a que se referia, (ii) linha clara que marcasse a estrutura do glifo, (iii) a inexistência ou existência mínima de pontos exteriores ao esqueleto e (iv) a possibilidade de divisão pelas partes anatómicas.

3.1.1 Interseção de retas aleatórias

Inicialmente optamos por testar a extração de esqueletos através da interseção de retas aleatórias com cada glifo. Para isso desenhamos um número pré-determinado de retas com posições aleatórias e determinamos os pontos de interseção (ou seja, pontos de entrada e de saída) das retas com cada glifo. Utilizando o par de pontos, é calculado o ponto médio (Figura 2) e o processo repete-se para todas as retas aleatórias desenhadas. Para a determinação do esqueleto são unidos todos os pontos médios resultantes (Figura 3).

Figura 2 — Demonstração do processo de cálculo dos pontos médios.

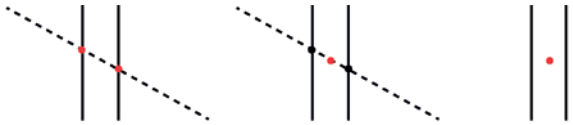

Figura 3 — Demonstração do processo de extração do esqueleto através da ligação dos pontos médios.

A Figura 4 apresenta gerações de pontos de esqueletos para todas as maiúsculas do alfabeto latino. Apercebemo-nos que ainda existiam alguns pontos de pontos de ruído, ou seja, pontos que idealmente não fariam parte do esqueleto de cada glifo. No entanto, a ligação dos pontos poderia diminuir esse problema.

Figura 4 — Pontos do esqueleto extraídos para todas as maiúsculas do alfabeto latino, utilização de 5000 retas.

Ligação de pontos — Do canto superior esquerdo ao inferior direito

Depois de detetados os pontos mais favoráveis tínhamos de uni-los para formar a linha do esqueleto. O primeiro método que desenvolvemos para unir os pontos resultantes foi através da ligação ao ponto à direita; quando não existia passava para baixo. No entanto, como os pontos resultantes eram ordenados da esquerda para a direita e de cima para baixo o resultado apresentava uma série de linhas horizontais que não pertenciam ao esqueleto, nem à forma (Figura 5).

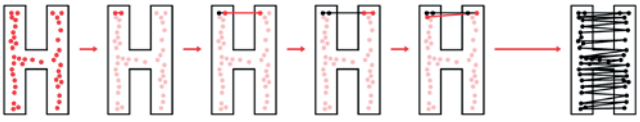

Figura 5 — Demonstração da ligação de pontos, primeira versão

Para eliminar as ligações inconvenientes acrescentamos uma verificação da distância entre os pares de pontos que seriam ligados. Na Figura 6 é possível observar os resultados gerados antes e depois da verificação da distância.

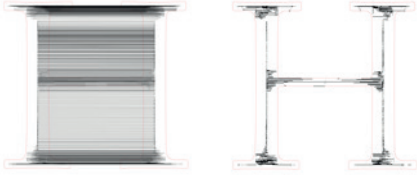

Figura 6 — Resultados antes e depois da verificação da distância, à esquerda e à direita respectivamente. Na segunda geração foi aplicada a distância máxima de 10% da altura do glifo.

Mais tarde, geramos esqueletos para todas as letras do alfabeto latino. Na Figura 7 estão os esqueletos gerados para as maiúsculas. Apesar dos resultados gerados apresentarem uma estética interessante e de ser possível identificar cada um dos caracteres, optamos por trocar o método de ligação de pontos. Tomamos esta decisão devido ao facto do esqueleto gerado ser, em grande parte, constituído por linhas horizontais, que na sua grande maioria não deveriam existir.

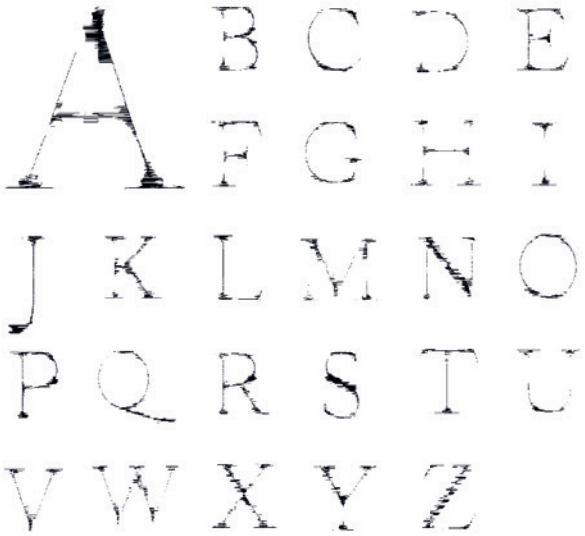

Figura 7 — Esqueletos gerados para todas as maiúsculas do alfabeto latino através do método da ligação pelos pontos ordenados

Ligação de pontos pelos mais próximos

Depois do método anterior não responder da melhor forma ao nosso objetivo decidimos implementar outra forma de ligação entre pontos. Para isso decidimos ordenar os pontos unindo sempre o mais próximo. Após algumas gerações apercebemo-nos que, tal como o que acontecia na função anterior, por vezes eram unidos pontos muito distantes. Isto acontecia quando o ponto que estávamos a unir situava-se uma posição sem saída e todos os pontos antes dele já estavam unidos. Por esse motivo foi também implementada uma verificação da distância entre os dois pontos a ser ligados.

Depois da melhoria no método geramos esqueletos para todas as letras do alfabeto latino. Na Figura 8 estão os esqueletos gerados para as letras maiúsculas. No entanto, os resultados apresentamos continuavam a não ser os mais promissores. Existiam demasiados pontos que não faziam parte do esqueleto ideal de cada glifo.

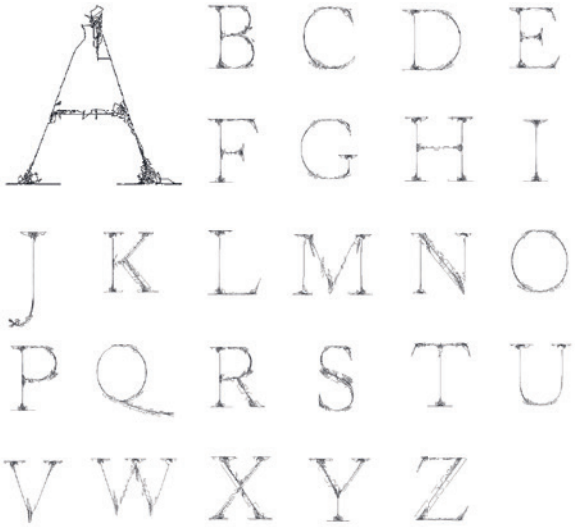

Figura 8 — Esqueletos gerados resultantes da ligação pelos pontos mais próximos. Foi aplicada a distância máxima de 10% da altura do glifo.

Ligação de pontos através de Regressões Lineares

Depois de duas funções desenvolvidas para ligar os pontos do esqueleto os resultados ainda não eram os mais favoráveis. Por esse motivo, optamos por arranjar outra forma de unir os pontos, tentando excluir os pontos irrelevantes. Decidimos então aplicar várias regressões lineares aos pontos médios resultantes da interseção das retas aleatórias com a forma.

Para contextualizar, uma regressão linear é uma equação que normalmente serve para descobrir um valor que não se consegue estimar inicialmente. No nosso caso, tornava-se útil pois os pontos de fora são excluídos já que a reta se aproxima mais do local onde exista uma maior quantidade de pontos. A Figura 9 demonstra os resultados a que chegamos.

Figura 9 — Resultado gerado depois da ligação de pontos através de regressões lineares. Aplicação de uma distância máxima de 5% da altura do glifo respectivamente.

Até ao momento, tínhamos desenvolvido três métodos para a ligação dos pontos e criação de esqueletos. No entanto, nenhum resultado parecia responder aos nossos objetivos. Possivelmente o problema não se devia à forma como estávamos a unir, mas ao facto de existirem demasiados pontos que não faziam parte do esqueleto ideal de cada glifo. O método de extração de pontos retornava uma conjunto de pontos que não fazia parte do esqueleto ideal. O grande problema era o acomular de pontos nas ligações entre as diferentes partes do esqueleto, decidimos então optar por desenvolver um outro método para a extração dos pontos do esqueleto.

3.1.2 Interseção de retas normais

O novo método que desenvolvemos consistia na determinação da reta normal para a curva num determinado ponto do contorno do esqueleto. Depois, era calculado o ponto médio resultante dos pontos de interseção da reta com a forma. Para contextualizar, a reta normal à curva num ponto é aquela que passa pelo ponto e é perpendicular à reta tangente da curva no ponto. Por outro lado, a reta tangente à curva num ponto é a aquela que toca em apenas um ponto. Na Figura 10 está a demonstração do método.

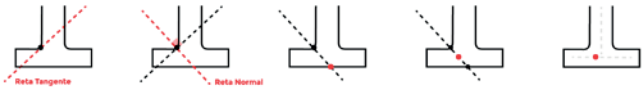

Figura 10 — Calculo da reta normal à curva num ponto e posterior cálculo do ponto médio.

Na sua essência, o sistema era semelhante ao método anterior de extração de pontos. Com a utilização do método anterior para a extração dos esqueletos surgem pontos que fazem parte do esqueleto. No entanto, como as retas são aleatórias existe também a possibilidade do surgimento de pontos que vão funcionar como ruído para a criação do esqueleto. Por outro lado, no método atual temos a vantagem de utilizarmos apenas o conjunto de retas mais promissoras, portanto possivelmente melhores resultados. A Figura 11 apresenta os pontos dos esqueletos gerados para todas as maiúsculas do alfabeto latino.

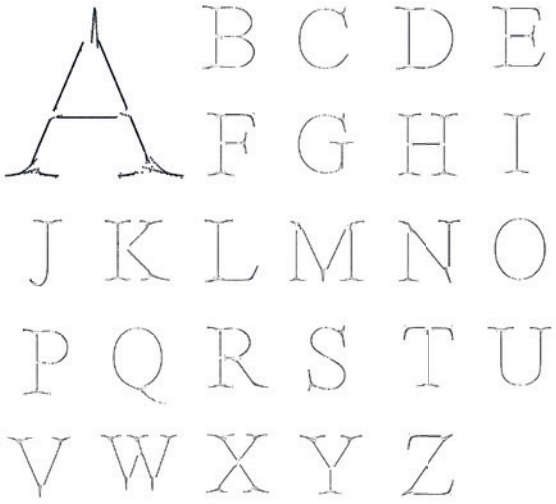

Figura 11 — Pontos do esqueleto gerados depois da aplicação do método da extração através do uso de retas normais.

Ligação de pontos pelos mais próximos

Uma vez mais, e agora com pontos mais promissores, tínhamos de ligar os pontos para formar o esqueleto. No método de extração anterior tinham sido criados uma série de métodos para a ligação de pontos. No entanto, como abordado anteriormente, não tiveram os resultados mais promissores devido ao conjunto de pontos recebido. Com a implementação desta nova forma de extração de pontos tínhamos o necessário para construir esqueletos sem ruído. Para validar este método testamos novamente todas as letras do alfabeto latino. Na Figura 12 estão apresentadas os esqueletos gerados para as letras maiúsculas. Após uma análise aos resultados gerados reconhecemos que este poderia ser o caminho deste projeto. No entanto, decidimos ainda testar a outra função que tínhamos desenvolvido anteriormente através do uso de regressões lineares.

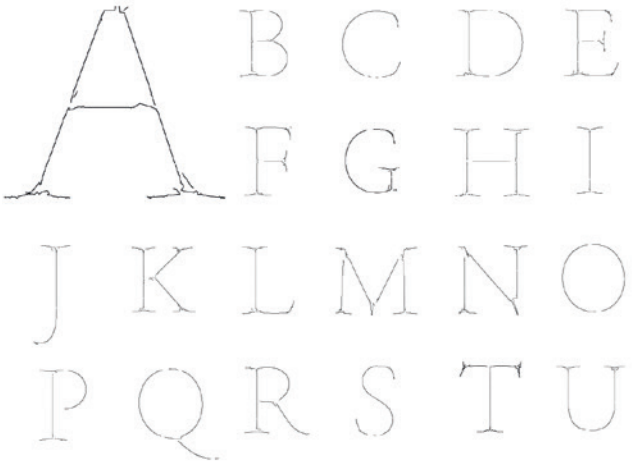

Figura 12 — Esqueletos gerados depois da aplicação do método de extração de pontos através de retas normais e o método de ligação de pontos mais próximos. Foi aplicada a distância máxima de 10% da altura do glifo.

Ligação de pontos através de Regressões Lineares

Este método de ligação foi criado para o método de extração de pontos anterior como tentativa de exclusão dos pontos irrelevantes. Por esse motivo, apenas serão apresentadas gerações resultantes da aplicação do método de regressões lineares (Figura 13) e a análise de resultados. Com a utilização deste método para a ligação dos pontos os resultados gerados são mais simplificados. No entanto, o esqueleto formado não parece ser completo, pois as várias linhas desenhadas não se ligam. Foi testada também a ligação dessas linhas, mas os resultados não eram promissores razão porque não estão aqui exemplificados. Em geral, o método de ligação anterior responde melhor aos nossos objetivos.

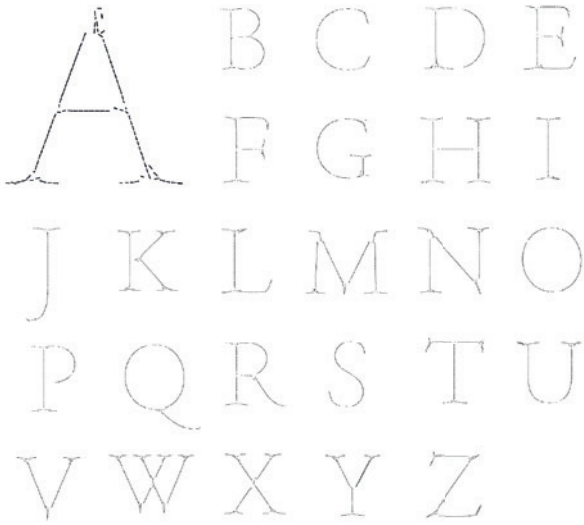

Figura 13 — Esqueletos gerados depois da aplicação do método de extração de pontos através de retas normais e o método de ligação de pontos através de regressões lineares. Foi aplicada a distância máxima de 10% da altura do glifo.

3.1.3 Técnica Final

Até ao momento tínhamos um sistema de extração de pontos mais ou menos satisfatório. No entanto, os esqueletos gerados ainda tinham alguns problemas por resolver. O sistema de ligação de pontos mais promissor era através da ligação por pontos mais próximos, mas mesmo esse, devido à grande quantidade de pontos, era constituído por linhas muito irregulares. Esse factor condicionava a validade dos esqueletos até agora gerados, pois não existiam linhas claras a marcar a estrutura do esqueleto.

Além disso, um dos nossos objetivos iniciais era dividir o esqueleto em partes anatómicas para posteriormente combinação de diferentes tipos de letra e ainda não estava feito. Refletimos sobre qual seria o próximo passo e encontramos um algoritmo que poderia ser implementado para facilitar o processo de extração de esqueletos. Mais tarde, apercebemo-nos também que, além dele, existiam uma série de métodos de extração de esqueletos, técnica bastante utilizada em domínios como a animação 3D (Schaefer, Yuksel, 2007; Tierny, Vandeborre, Daoudi, 2008; Tierny, Vandeborre, Daoudi, 2006).

O algoritmo que encontramos, denominado *Zhang-Suen Thinning Algorithm* (Wu, Marquez, 2003), tem como objetivo extrair as linhas estruturais de uma imagem binária. Por outras palavras, o algoritmo recebe uma imagem constituída apenas por *pixels* de duas cores diferentes, por exemplo, preto e branco e devolve-a modificada. Todos os *pixels* imprescindíveis para a compreensão da mesma são retirados, portanto é ideal para o nosso caso, a extração de esqueletos. Encontrada a possível solução para a extração dos esqueletos, precisava de ser validada para o nosso problema. Este método difere muito dos anteriores pelo facto de não separar a extração de pontos com a ligação entre eles, é logo extraída a estrutura do glifo.

O processo de geração do esqueleto de cada glifo inicia-se com o varrimento de todos os *pixels*. A cada iteração são eliminados os *pixels*

que se encontrem mais afastados do esqueleto. Só quando nenhum *pixel* sofrer alteração é que o processo é dado como terminado. Na Figura 14 é mostrado o esqueleto gerado para um 'h'.

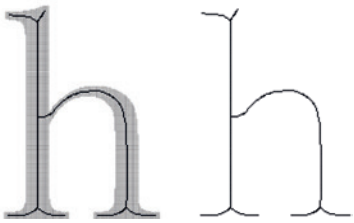

Figura 14 — Pixels iniciais e esqueleto final gerado depois da aplicação do Método Final de Extração de Esqueletos

De modo a validar o método testamos com vários glifos, na Figura 15 estão os esqueletos gerados para as letras maiúsculas do alfabeto latino. Como previsto, este método de extração de esqueletos apresentou resultados melhores que os métodos anteriores.

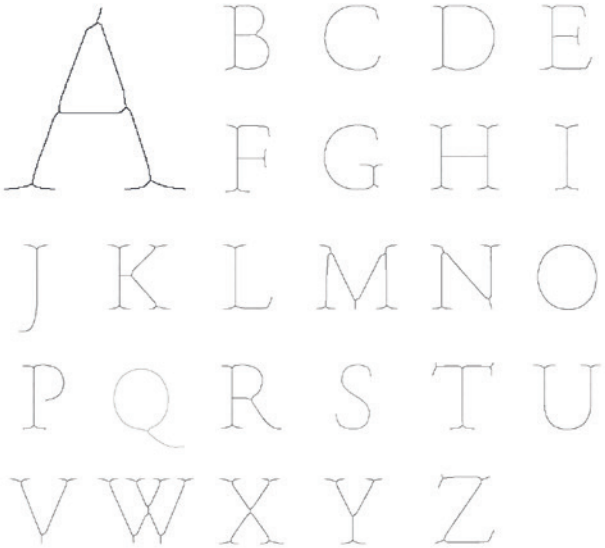

Figura 15 — Esqueletos gerados depois da aplicação do Método Final de Extração de Esqueletos

Divisão do esqueleto em diferentes partes anatómicas

Depois do sucesso do último método da extração de esqueletos, com a implementação do algoritmo de *Zhang-Suen*, tínhamos finalmente todas as condições para a divisão do esqueleto em diferentes partes anatómicas. Após uma análise dos esqueletos gerados reparamos que quando um ponto fazia parte de três segmentos ele dividia diferentes partes do esqueleto. De forma a testar essa teoria percorremos todos os pontos do esqueleto e quando era encontrado um ponto de fronteira era criado uma nova linha (Figura 16).

Figura 16 — Esqueletos gerados depois da aplicação da Técnica final de extração de esqueletos com a divisão em partes. Os círculos destacam os pontos de fronteira. Foi aplicada uma cor diferente para diferenciar as diferentes partes.

Simplificação de pontos

Ao longo do processo, fomo-nos apercebendo que por vezes os esqueletos gerados tinham algumas irregularidades. A ligação dos pontos era, por vezes, em zig-zag e portanto causava algum ruído. De modo a diminuir os pontos em excesso e tornar a estrutura dos esqueletos mais claros decidimos fazer a implementação de um outro algoritmo, *Ramer Douglas Peucker* (Wu, Marquez, 2003). Dada uma curva composta por segmentos de reta ele devolve uma curva semelhante com menos pontos. O algoritmo verifica se a curva é semelhante através do cálculo da distância máxima entre a curva original e a simplificada.

O método de simplificação foi então aplicado a cada um dos segmentos dos esqueletos gerados (Figura 17). Os resultados são interessantes no ponto em que permitem grande diversidade. O uso de um *input* que possa modificar o grau de simplificação dos esqueletos poderia gerar glifos bastante diversificados.

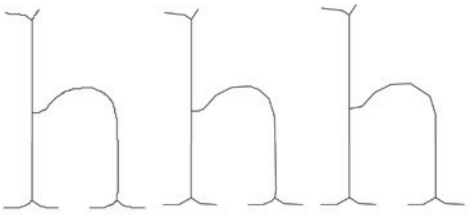

Figura 17 — Esqueletos gerados depois do uso de diferentes graus de simplificação (0.5,1,2)

3.2. Cruzamento de esqueletos

Para o nosso projeto pretendíamos também que a estrutura dos tipos gerados fosse totalmente automática. Um outro ponto relevante era a possibilidade dos tipos gerados se poderem adaptar a diferentes circunstâncias. Portanto, a combinação de partes anatómicas oriundas de um conjunto diversificado de tipos de letra traria um vasto conjunto de possibilidades para a estrutura das fontes geradas. Para avaliar se as partes entre diferentes esqueletos eram propícias a serem associadas determinamos a velocidade angular e o ponto central para cada uma delas. Desse modo, cada uma das partes não precisava de ser totalmente igual a outra para ser associada.

Para contextualizar, a velocidade angular é um vetor que representa o processo da mudança de orientação de uma dada linha. Para uma reta o valor será igual a 0, este vai aumentando consoante esta se torna mais curva. O ponto central é uma média de todos os pontos da parte em questão. Com o cálculo destas duas variáveis podemos identificar uma parte de um caractere.

Depois, comparamos partes entre glifos. Quando as partes comparadas tinham uma velocidade angular e um ponto central semelhante eram consideradas correspondentes. Para demonstrar a Figura 18 apresenta combinações de partes para uma série de caracteres.

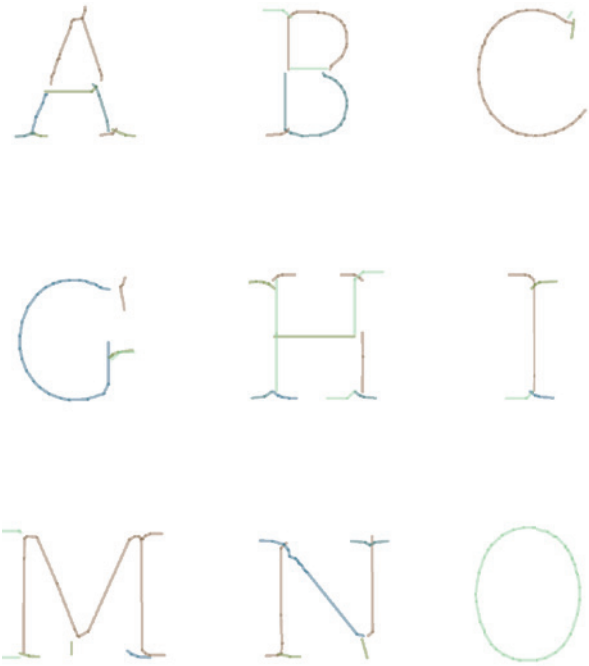

Figura 18 — Esqueletos gerados através da combinação aleatória de três tipos de letra (cada cor representa um tipo de letra)

Ao analisar os resultados reparamos que nenhuma das combinações criadas torna o esqueleto ilegível, apesar de por vezes as linhas não se unirem completamente. Esse fator é devido ao uso de vários tipos de letra com medidas diferentes. No geral, ficamos satisfeitos com o resultado pois respondia aos nossos objetivos para o esqueleto.

3.3. Dar corpo ao esqueleto

Até agora, tinham sido estabelecidos esqueletos para diferentes tipos de letra. Tinham sido combinados e, como consequência, tínhamos criado estruturas para os tipos de letra que o nosso sistema iria gerar. Determinadas todas estas variáveis faltava dar corpo aos esqueletos gerados. Anteriormente tínhamos referido que o processo de extração se iniciava com a exclusão dos pixels mais afastados do centro da forma. Portanto, ao calcular a distância do pixel do background mais próximo até ao pixel do esqueleto final determinamos a largura do tipo de letra original. Com essa medida podemos replicar o tipo, ou aumentar ou diminuir o peso de forma proporcional (Figura 19). Para testar a ideia de replicação dos tipos iniciais guardamos a largura de cada tipo de letra, em cada ponto. Fizemos este processo para três tipos de letra e no final combinamos as partes de forma aleatória. Utilizando círculos percorremos as partes do esqueleto final utilizando a largura guardada. Para cada parte, corremos ponto a ponto e aumentamos ou diminuímos gradualmente o raio do círculo desenhado (Figura 20).

Figura 19 — Método de preenchimento dos pontos intermédios

Figura 20 — Glifos gerados como tentativa de replicação dos glifos que serviram como input para a extração do esqueleto. Foram usadas partes de três tipos de letra diferentes e estas foram escolhidas aleatoriamente.

Os resultados obtidos eram interessantes, pois, além das irregularidades, era possível observar características dos tipos de letra utilizados. No entanto, como definido anteriormente o nosso objetivo era construir tipos de letra que fossem constituídos por camadas. Ainda utilizando os círculos, geramos vários caracteres utilizando cores diversificadas e a transparência em nosso proveito. De seguida são apresentados uma série de resultados. Em todas as gerações (Figura 21, 22 e 23) o preenchimento da forma é acompanhado pelo esqueleto. As cores utilizadas foram geradas de forma aleatória, associando uma cor a cada tipo de letra. A utilização de transparências permite uma melhor combinação entre as diferentes camadas.

Figura 21 — Glifos gerados com a combinação de quatro tipos de letra diferentes. Aplicação de cores diferentes para cada tipo de letra.

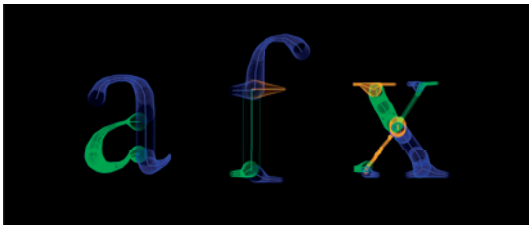

Figura 22 — Glifos gerados com a combinação de quatro tipos de letra diferentes. Aplicação de cores diferentes para cada tipo de letra.

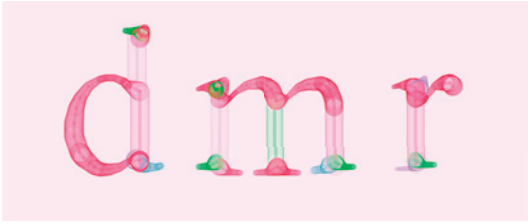

Figura 23 — Glifos gerados com a combinação de quatro tipos de letra diferentes. Aplicação de cores diferentes para cada tipo de letra.

No seguimento das experimentações, decidimos utilizar outros símbolos para a geração de glifos. Nesta geração, Figura 24, optamos por utilizar o quadrado rodado em vez do círculo. Decidimos também deixar o esqueleto de parte e as transparências e alterar o número de módulos desenhados em cada parte anatómica. Com a diminuição do número de módulos a serem desenhados a variação da densidade entre as partes do glifo tornou-se mais visível. Isto acontecia, porque todas as partes eram desenhadas com o mesmo número. Portanto, nas partes mais curtas existia uma área de maior acomulação de cor. Apesar de não ter sido previsto à partida essa variação da densidade tornava-se interessante no ponto que tornava cada parte mais única.

Figura 24 — Glifos gerados com a combinação de quatro tipos de letra diferentes, sem esqueleto. Utilização do quadrado rodado como módulo.

Mais tarde, geramos também glifos com a utilização de diferentes módulos para preenchimento das camadas. Atribuímos um módulo para cada tipo de letra (quadrado rodado, triângulo retângulo, círculo e cruz). Para esta geração, Figura 25, atribuímos uma cor aleatória, além do módulo para cada tipo de letra. A utilização de duas variáveis como representação do tipo de letra ajudam à sua diferenciação. Além disso, a combinação dos diferentes módulos apenas com contorno acrescenta detalhes aos glifos obtidos.

Figura 25 — Glifos gerados com a combinação de quatro tipos de letra diferentes. Utilização de vários módulos e diferentes cores para cada tipo de letra.

Mais tarde, optamos por uma opção inspirada em técnicas de caligrafia. Para isso decidimos aplicar uma rotação aos módulos desenhados. Deste modo, o desenho de cada elemento era rodado consoante a inclinação da linha do esqueleto. Aplicando diferentes ângulos de rotação chegamos às gerações das Figuras 26 e 27.

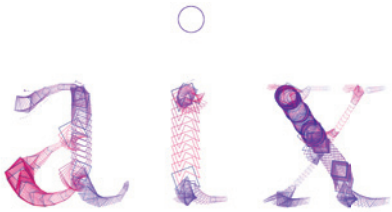

Figura 26 — Glifos gerados com a combinação de quatro tipos de letra diferentes. Utilização de vários módulos, cores e aplicação de rotação.

Figura 27 — Glifos gerados com a combinação de quatro tipos de letra diferentes. Utilização de vários módulos e rotação.

Mais tarde, decidimos fazer algumas experiências modificando a largura dos glifos. Para a geração da Figura 28 descartamos a largura do glifo original. Por outras palavras, determinamos uma largura fixa para todos os módulos (partes anatómicas).

Figura 28 — Glifos gerados com a combinação de quatro tipos de letra diferentes. Utilização de vários módulos e aplicação de rotação e largura do módulo fixa.

Por outro lado, testamos também o uso da largura dos tipos de letra iniciais para alteração do peso dos tipos gerados. Na Figura 29 é apresentada uma geração de glifos com a diminuição da largura relativamente aos originais, são utilizados módulos com apenas contorno e preenchidos. Testamos também a geração de glifos com um peso superior ao original (Figura 30).

Figura 29 — Glifos gerados com a combinação de quatro tipos de letra diferentes. Utilização de vários módulos. Aplicação de rotação e peso inferior aos tipos originais.

Figura 30 — Glifos gerados com a combinação de quatro tipos de letra diferentes. Utilização de vários módulos. Aplicação de rotação e peso superior aos tipos originais.

Depois destas gerações, concluímos que a variação da largura dos tipos poderá ser uma variável para estar à escuta na aplicação deste sistema no mundo real. No geral, todas estas gerações e todas aquelas que podem ser geradas no nosso sistema são possibilidades. Não existe uma versão mais correta do que a outra.

4. Conclusão e trabalho futuro

Neste projeto desenvolvemos um sistema que cria tipos de letra de forma automática. O processo de geração inicia-se com o desenvolvimento da estrutura dos tipos e codificação dos diferentes elementos da anatomia da letra em diversas camadas. Os glifos criados são resultado da combinação de camadas de diferentes tipos de letra e aplicação de um método de preenchimento das mesmas.

Em retrospectiva, o sistema final responde aos objetivos propostos inicialmente. Os glifos gerados são percetíveis, mas obviamente não são a melhor opção para texto longo devido ao seu detalhe. Contudo, os tipos de letra gerados podem facilmente ser adaptamos para um logótipo ou

uma identidade dinâmica. De facto, existe uma procura emergente por identidades visuais baseadas em formas de letras projetadas especificamente para elas e, portanto, o nosso sistema pode responder a essas aspirações. Desta forma, somos capazes de criar inúmeras variações para os logótipos do mesmo conceito.

Apesar de o sistema responder aos nossos objetivos iniciais seria interessante melhorar alguns aspetos. Para que os tipos de letra gerados estejam de acordo com as regras tradicionais de desenho de tipos, as partes comuns entre caracteres do mesmo tipo teriam de ser iguais (exemplo: o 'p' e o 'q' terem sempre o mesmo descendente). No entanto, no nosso sistema tal ainda não acontece, mas pensamos não ser um problema muito difícil de resolver como trabalho futuro. Além disso, poderíamos também melhorar um dos métodos descartados que era a deteção de serifas e aplicá-lo ao novo sistema. Esta alteração tornaria o sistema mais completo, permitindo ao utilizador escolher a geração de um tipo serifado ou não serifado.

Seria também interessante a projeção do sistema em entidades reais, como a identidade de um museu ou de uma conferência e observar de que forma a sua componente dinâmica se poderia adaptar aos dados recebidos.

5. Referências

PARENTE, Jéssica — ***Design Generativo de Tipos de Letra***. Dissertação de Mestrado. Coimbra: Universidade de Coimbra, 2018

CHENG, K. — ***Designing type.*** Laurence King Publishing, 2006.

SCHAUGHNESSY, A; BIERUT, M. — ***Graphic design: a user's manual***. Laurence King, 2009

LEHNI, J. — ***Typeface As Programme***. 2011 [Consult.20 maio 2018]. Disponível na internet:. <https://www.typotheque.com/articles/typeface_as_programme>

LUPTON, E. — ***Pensar com tipos: guia para designers, escritores, editores e estudantes***. Cosac & Naify, 2006

FONTFONT — ***FF Beowolf***. n.d.a [Consult.22 maio 2018]. Disponível na internet:. <https://www.fontshop.com/families/ff-beowolf>

MAINZ, G. M. — ***Gestalten mit Code, FH Mainz***. n.d.a Consult.22 maio 2018]. Disponível na internet:. <http://generative-typografie.de/generativetypografie/bastard/>

MAINZ, G. M. — ***Gestalten mit Code, FH Mainz.*** n.d.b Consult.22 maio 2018]. Disponível na internet:. <http://generative-typografie.de/generativetypografie/fontmix/>

FONTFONT — ***FF ThreeSix.*** n.d.a [Consult.21 maio 2018]. Disponível na internet:. < https://www.myfonts.com/fonts/fontfont/threesix/>

MUIRMCNEIL — **ThreeSix**. n.d.a [Consult.21 maio 2018]. Disponível na internet:. <http://www. muirmcneil.com/project/threesix-3/>

MUIRMCNEIL — **Intersect**. n.d.b [Consult.21 maio 2018]. Disponível na internet:. <http://www. muirmcneil.com/project/intersect-2/?section=about>

SILANTEVA, D. — **Typographic Music**. 2011 [Consult.22 maio 2018]. Disponível na internet:. <http://www.ddina.com/index.php?/2011/typographic-music/2/>

SULLIVAN, M. — **Dina Silanteva —Typographic Music.** 2011 [Consult.22 maio 2018]. Disponível na internet:. <http://www.typetoken.net/publication/dina-silanteva-—-typographic-music/>