

Experimentações com design generativo no desenvolvimento de padronagens para produtos de moda

Experiments with generative design in the development of seamless patterns for fashion products

COSTA, Diva Lúcia Vieira; Mestranda; Universidade Federal do Rio de Janeiro
divalucosta@gmail.com

ESPERANÇA, Claudio; Doutor; Universidade Federal do Rio de Janeiro
esperanc@cos.ufrj.br

KOSMINSKY, Doris; Doutora; Universidade Federal do Rio de Janeiro
doriskos@eba.ufrj.br

MOURTHÉ, Claudia; Doutora; Universidade Federal do Rio de Janeiro
claudiamourthe@eba.ufrj.br

Este artigo aborda diferentes formas de experimentação do design generativo em projetos de design de superfície para a moda no contexto da indústria 4.0. O objetivo é propor novas maneiras de desenvolvimento e visualização de padronagens por meio da interação com o design generativo, considerando a participação do usuário em um processo de co-criação. Com a evolução do design e da indústria, a customização em massa possibilita a personalização de produtos dentro de uma cadeia produtiva. E com a participação do usuário na co-criação de novos designs, a experiência ultrapassa as barreiras do consumo e o consumidor poderá participar do processo de criação de padronagens generativas, introduzindo características provenientes do seu próprio sistema cultural, visualmente percebidas por meio das formas e cores.

Palavras-chave: Design generativo; Padronagem; Design de moda.

This article presents different ways of experimenting with generative design in surface design for fashion projects in the context of Industry 4.0. The aim is to propose new ways of developing and visualizing patterns through interaction with generative design, considering the user's participation in a co-creation process. With the evolution of design and industry, mass customization makes it possible to customize products within a production chain. And with the user's participation in the co-creation of new designs, the experience goes beyond consumption barriers and the consumer will be able to participate in the process of creating generative patterns, introducing characteristics from their own cultural system, visually perceived through the forms and colors.

Keywords: Generative design; Pattern design; Fashion design.

1 Introdução

Este artigo tem como objetivo apresentar o design generativo na criação de padronagens por meio da linguagem de programação JavaScript utilizando a biblioteca p5.js. Na primeira parte deste estudo são apresentados conceitos relacionados à evolução do design como justificativa para a inserção do design generativo nos processos de desenvolvimento de produtos.

Na segunda parte é apresentada a customização em massa, contextualizada diante do cenário da indústria 4.0 e da necessidade de personalização dos produtos, destacando preferências individuais. São apresentados alguns casos de empresas e instituições que já realizam projetos de personalização de produtos no contexto industrial.

Na terceira parte são estabelecidas relações entre o design generativo e o design de superfície, utilizando o desenvolvimento de padronagens como recorte para a experimentação prática. A padronagem possui particularidades que são embasadas por conceitos do design de superfície. A aplicação prática é realizada por meio de experimentações, desenvolvidas na plataforma p5.js, que permite a utilização de códigos para gerar formas, cores e outros elementos, viabilizando a construção de padronagens generativas.

Este estudo pretende alavancar discussões acerca do design generativo e sua utilização para o desenvolvimento de projetos em design de moda. A participação do usuário no processo de criação de novos produtos de moda é referenciada de maneira breve neste artigo, porém faz parte de um estudo maior que está sendo desenvolvido como tema de mestrado da autora.

2 Metodologia

A pesquisa realizada para o desenvolvimento do artigo é qualitativa, de caráter exploratório e de natureza aplicada. O método utilizado se baseia na revisão bibliográfica em periódicos, artigos, livros e vídeos relacionados com os temas do estudo.

Foram abordados na pesquisa referenciais teóricos relacionados ao design generativo, design de superfície, indústria 4.0, customização em massa, co-criação e design de moda. Foram citados casos de aplicação dos conceitos abordados no mercado de moda e na indústria. E foram realizadas análises a partir das observações via pesquisa em sites e em documentos de projetos implementados, como no caso da planta de confecção 4.0 do SENAI CETIQT. Também foram realizadas experimentações utilizando o p5.js para apresentação das possibilidades existentes no design generativo e que podem ser aplicadas ao desenvolvimento de padronagens generativas.

3 O design generativo: uma evolução do design

Com a dinâmica evolução tecnológica e a fusão entre o mundo físico e digital, o papel do designer está mudando. Em resposta a essas mudanças, as práticas já estabelecidas no campo do design podem não ser suficientes considerando que o design vai além apenas dos desejos de consumo e da concretização de ideias (MCCORMACK et al., 2004). Neste contexto, o artefato é reconsiderado na visão do designer de produto, deixando de ser estático e passando a ter mais relevância nas ações que o manipulam, os sistemas e os processos que interagem entre si e com usuários, gerando novos resultados, novas interações, novos artefatos.

O design generativo é baseado em um conjunto de regras ou algoritmos que irão produzir um resultado, uma forma, uma cor, um movimento, ou todos ao mesmo tempo, considerando características individuais (MALIK, 2016). Assim como na natureza, o design generativo é

desenvolvido de dentro para fora, baseado no conjunto de regras e variáveis, para então apresentar um resultado. A partir da introdução dessas regras, é possível considerar aspectos culturais, que irão interferir no resultado gerado. Malik (2016) faz uma analogia com um floco de neve, que surge a partir de condições do ambiente para que seja formado, tornando assim cada floco de neve único. Ou seja, as condições e características do ambiente influenciam no resultado. E ao colocar um novo produto no mercado, as condições que o cercam devem ser consideradas em etapas anteriores, na criação e no desenvolvimento de produto.

O mecanismo do design generativo é artificial, pois é feito pelo ser humano, mas se baseia em características naturais ou ao menos na simulação de características naturais. A linguagem, o processo de design e os parâmetros inseridos no sistema são universais, amplificando seu alcance global, não sendo necessário o domínio de habilidades culturais ou relativas em um primeiro momento, pois um mesmo conjunto de códigos é capaz de gerar diferentes resultados ao interagir com diferentes pessoas, que possuem características individuais, particulares.

Sistemas generativos estão intimamente ligados ao conceito de síntese que, segundo significado da palavra, refere-se à operação de corpos simples que se reúnem para formar corpos compostos, ou conjunto de compostos que se reúnem para formar algo mais complexo ainda. Esses conceitos estão presentes na própria natureza e em sistemas naturais, os quais são verdadeiras inspirações para a programação computacional. A diversidade e adaptabilidade da vida na Terra demonstra a característica evolucionária, ou seja, o potencial dos mecanismos naturais para superar problemas de projeto e gerar soluções, gerar novas ideias e propagar a diversidade a partir de unidades relativamente simples (MCCORMACK, et al., 2004). Assim como a natureza, “o design generativo funciona como um sistema que nos oferece infinitas variações de um produto” (MALIK, 2016). Os sistemas generativos são capazes de gerar formas e padrões complexos a partir de especificações simples, assim como os sistemas naturais, ou seja, do mais simples ao mais complexo, usando os mesmos itens, gerando inúmeros resultados possíveis, adequados a cada situação específica.

A partir desta característica evolucionária, o design generativo pode ser definido por meio de procedimentos adaptativos, condicionados e relacionados a um efeito, de acordo com a condição estabelecida e que pode ser transformada por outros públicos por meio da sua própria interpretação do artefato em questão. Ou seja, qualquer pessoa poderá desempenhar um papel no design de produto (MALIK, 2016). O design natural é baseado na individualidade, pois assim como na natureza, não há duas flores iguais, duas nuvens iguais, duas zebras iguais. Ao inserir aspectos individuais dos consumidores em um projeto de design, possibilita a quebra da homogeneidade cultural de consumo. Essa ruptura evidencia as escolhas e características individuais das pessoas, o que poderá alimentar o desenvolvimento de novos produtos.

Os sistemas generativos oferecem não só uma maneira de produzir um artefato dinâmico, mas oferecem também uma metodologia e uma filosofia para ver o mundo em termos de processos dinâmicos e seus resultados (MALIK, 2016). Segundo McCormack et al. (2004), o design generativo oferece novos modos de experiência estética, ou seja, a experiência é o foco principal e não apenas o produto final, resultado de inúmeros processos conhecidos pelo designer, mas que o usuário não se envolve até tê-lo pronto em suas mãos. O projeto de design generativo se torna um meta-design, no qual o foco na experiência traz um interesse para essa pesquisa, baseada na co-criação, ou seja, do resultado da interação entre designer e usuário. E isso só se torna possível pelo mapeamento de processos e parâmetros nos momentos iniciais no desenvolvimento de produto, ação feita pelo designer com a colaboração e participação de usuários. Considerando a interação do consumidor, alguém que

não é designer e participa do processo de criação, poderá realizar contribuições com sua visão única do sistema cultural em que está inserido. A partir desses dados, o designer poderá definir os parâmetros que serão utilizados na programação computacional.

Essas características do projeto são trazidas para o design generativo por meio da customização em massa, na qual consumidores podem trazer variações nos designs, democratizando o design e sua fabricação, permitindo que os consumidores utilizem o conjunto de regras estabelecidas pelos designers para projetar seus próprios produtos.

4 Customização em massa

A partir de um estudo realizado por Fralix (2001) a customização em massa vem crescendo, pois une o melhor da era artesanal e da era industrial. A primeira, quando produtos eram fabricados de acordo com as especificações dos consumidores, porém apenas parte da população tinha condições financeiras de obter o produto. E a segunda, na qual todos poderiam obter o mesmo produto porque era acessível financeiramente, mas eram produtos idênticos, sem diferenças individuais. A customização em massa permite personalizar produtos em um sistema de produção industrial, com preços mais acessíveis do que o produto artesanal e com diferenças individuais, desconsideradas no sistema industrial tradicional.

A customização em massa não exclui os produtos feitos sob medida, pois possui outras características para sua criação e produção, considerando equipamentos de alta performance, com scaners corporais e impressão digital. Segundo Joe Pine (apud FRALIX, 2001, tradução nossa) “o objetivo da customização em massa é fornecer variedade suficiente para que os desejos do consumidor sejam satisfeitos”. Porém, ao mesmo tempo, atender a uma multiplicidade de demandas pode resultar em uma produção artificial em massa, o que pode gerar uma insatisfação dos consumidores (MALIK, 2016), sendo o oposto do que trata a customização em massa, na qual a experiência de compra e a satisfação são aspectos importantes. Pine (ibidem) reforça que a variedade em si não é customização e que os recursos personalizáveis devem incluir aspectos que o consumidor considera relevantes. Por isso, a participação do usuário no processo de desenvolvimento de produto, a co-criação, se faz importante para identificar quais são esses aspectos. Além disso, os produtos personalizados não devem necessariamente custar mais, apesar do investimento inicial em tecnologia que viabilize sua produção.

Com a evolução da tecnologia, principalmente em meios digitais, é possível reconfigurar as características do produto de forma rápida para atender às necessidades do consumidor, algo que em 2001, na época do lançamento do artigo de Fralix, era um problema levantado por ele. Mas, com sistemas computadorizados e outras tecnologias da indústria 4.0, como a tecnologia da informação e a automação, é possível criar condições que permitam a interação com consumidores e também viabilizar a produção destes produtos (FRALIX, 2001; BRUNO, 2017).

4.1 Participação do usuário da indústria da moda

O design participativo, ou co-criação, permite que usuários, ou seja, pessoas que não são designers, participem do processo criativo de um produto. Essa participação é realizada por meio da colaboração de informações sobre determinados produtos ou ocasiões de uso de determinados produtos de moda. Os usuários são reconhecidos como especialistas e podem contribuir com pontos de vista específicos de quem utiliza o produto em diferentes situações e contextos.

Considerando que atualmente os produtos de moda são criados por equipes de designers que, por muitas vezes, se utilizam unicamente das tendências de moda divulgadas por sites

especializados e redes sociais, os produtos produzidos por essas equipes poderão se tornar obsoletos em um curto espaço de tempo. Compreender aspectos relevantes sobre diferentes contextos em que a moda atua, com a visão interna, de quem realmente vive esses contextos, é um aspecto importante para a co-criação de novos produtos. Dessa forma, por meio da interação com o usuário nas etapas de criação e desenvolvimento de produtos, poderão ser considerados outros aspectos além das tendências de moda unicamente.

A partir do design participativo, nas etapas de criação e desenvolvimento de novos produtos, designers poderão utilizar as informações dos usuários e integrá-las em sistemas computacionais para gerar diferentes resultados. E em etapas posteriores, no ponto de venda, com o design generativo e sistemas digitais que possibilitam a interação com usuários, a experiência de compra poderá ser transformada em um momento de personalização do produto, e até mesmo de entretenimento, diante de opções disponíveis previamente programadas para o momento de interação. O consumidor irá ser o “criador” de um produto personalizado para ele, adequado a suas expectativas e desejos.

A vantagem de ter o usuário participando do processo de co-criação do produto é poder produzir somente aquilo que o consumidor deseja, sob demanda, ao invés de produzir quantidades excedentes e que podem não agradar totalmente ao público, acabarem aumentando o estoque e, consequentemente, o desperdício e o próprio lucro das empresas. A produção sob demanda evita a produção excessiva, pois está voltada para o atendimento de necessidades imediatas do usuário. Em um sistema de produção tradicional, em que centenas de peças são produzidas, se consideradas as tendências de moda e desejos momentâneos dos consumidores como únicas fontes de informação no processo de desenvolvimento de produtos, o excesso de produção que não é absorvida pelo mercado consumidor poderá se tornar resíduo em um curto espaço de tempo.

4.1.1 A indústria 4.0 na implementação de soluções customizadas

Com o avanço da tecnologia, a aplicação dos conceitos da indústria 4.0 tornam possível a aproximação de consumidores com as cadeias produtivas, por meio da customização de massa no vestuário (BRUNO, 2017). Segundo Bruno, desde 2012, alguns autores já identificaram a manufatura social como um novo modo de fabricação que permite a qualquer pessoa participar do processo de manufatura. A manufatura social é uma nova maneira de introduzir a customização de massa na indústria de confecção a partir de tecnologias emergentes, como o espelho 3D de virtualização da prova de roupas. Neste modelo, o consumidor é completamente envolvido no processo produtivo ao definir suas preferências a partir de parâmetros pré-estabelecidos, como cor, estampa e forma, por exemplo. O consumidor inicia a produção de um produto em uma fábrica inteligente, criada para realizar um processo enxuto de fabricação, reduzindo custos e aumentando a satisfação do consumidor.

As minifábricas são modelos de manufatura que utilizam de tecnologia inovadora, como robótica, sistemas de processamento de imagens de alta precisão, comunicação entre máquinas e grande velocidade para manipular tecidos e criar roupas de alta qualidade e com custo mais baixo do que a produção feita em outros países (BRUNO, 2017, p.44). As minifábricas possuem um projeto modular que permite sua mobilidade, baixo impacto ambiental e ainda empregam estratégias que possibilitam a conexão direta com o consumidor, que ativa a produção pela compra. Segundo Vima (2015, apud BRUNO, 2017), a minifábrica permite ainda uma “lucratividade superior à de uma abordagem típica de produção de massa da cadeia de valor global”, viabilizando a fabricação de produtos personalizados em um sistema industrial.

A customização em massa já é uma realidade em empresas como Nike e Adidas, que criaram linhas de produtos customizados por clientes, como casacos e tênis, produzidos a partir da escolha de características personalizadas. Em ambos os casos, as empresas contam com sistemas digitais interativos para possibilitar o consumidor visualizar e compreender como fazer a personalização do produto.

A planta de confecção 4.0, instalada nas dependências da faculdade SENAI CETIQT (Serviço Nacional de Aprendizagem Industrial – Centro de Tecnologia da Indústria Química e Têxtil) é um exemplo de modelo produtivo que utiliza a realidade aumentada para a visualização do produto criado pelo consumidor. A planta é um protótipo de um sistema industrial, instalada para demonstrar a aplicação dos princípios e conceitos da indústria 4.0 na indústria têxtil e de confecção. E semelhante à Nike e Adidas, o consumidor escolhe as características do produto a partir das opções disponíveis, porém, neste caso, em um espelho virtual, que projeta as opções selecionadas em uma visualização em realidade aumentada, sobre a imagem da pessoa, em tempo real. Após realizar suas escolhas, o consumidor faz a sua “compra”, iniciando a produção do seu produto personalizado, fabricado com máquinas industriais, com processos integrados e automatizados. E em menos de 30 minutos, o consumidor tem o seu produto pronto.

As possibilidades de inserir o consumidor em processos de co-criação de novos produtos de moda é algo possível devido às tecnologias da indústria 4.0. Por meio de protótipos digitais ou até mesmo da realidade aumentada, o consumidor pode visualizar o produto sendo customizado em tempo real, visualizando as suas escolhas de cores e estampas, por exemplo. E o design generativo é capaz de integrar esses processos ao permitir a criação de inúmeras possibilidades a partir da seleção de características programadas pelo designer, se revelando como uma evolução do design, inclusive no desenvolvimento de produtos de moda.

5 O design generativo na criação de produtos de moda

A partir dos exemplos citados, é possível considerar que a experiência em personalizar um produto e produzi-lo de imediato, é algo inovador e até mesmo fora do comum, sob a perspectiva da indústria da moda, que produz milhares de produtos em seu sistema de produção tradicional e distribui ao redor do mundo, muitas vezes sem considerar as características individuais e dos sistemas culturais em que o público está inserido. Poder programar parâmetros e condições para a personalização de produtos pelo consumidor, permite criar experiências além do consumo. Esses momentos de interação com o consumidor podem ser considerados relevantes, nos quais as pessoas buscam por experiências diferenciadas, até mesmo de entretenimento, e que agreguem mais ao seu próprio estilo de vida e suas preferências individuais.

Para criar opções de personalização de produtos, deve-se considerar as características individuais de futuros consumidores, integrantes de seus sistemas culturais. O mapeamento de características individuais se faz necessário para poder incluir seus aspectos na co-criação de produtos de moda, desenvolvidos por designer e com a participação dos usuários.

No contexto de co-criação, no qual o consumidor interage com as opções de personalização do produto, percebem-se dois momentos de criação: o primeiro momento sendo aquele em que o designer estabelece os parâmetros do produto, prevendo possíveis desejos e resultados da interação com o seu público; e o segundo momento, aquele em que o consumidor usa esses parâmetros para combinar atributos e gerar o seu resultado, um produto personalizado.

No primeiro momento, o designer precisa se basear em algum dado para definir os parâmetros dos produtos. Relatórios de pesquisa de tendências de comportamento e de moda podem ser

usados, pois trazem uma gama de informações sobre pessoas e aspectos técnicos de produtos de moda, como cores, formas e estampas. Com a definição inicial dos parâmetros, o designer já consegue visualizar resultados possíveis dentro de uma regra ou condicional estabelecida por ele. Porém o designer não pode assegurar quais serão os resultados gerados na interação com o consumidor. Ou seja, o designer não tem total domínio do resultado a ser apresentado, apenas pressupõe quais as possibilidades poderão ser geradas a partir de inúmeras combinações possíveis entre os atributos considerados, neste caso, forma, cor e estampa.

No segundo momento, a partir dos atributos disponibilizados para o consumidor, este poderá gerar resultados que poderão ser os mesmos dos previstos pelo designer, ou outros resultados que irão surpreender até mesmo quem projetou. Neste momento, o consumidor carrega toda a sua bagagem cultural do momento em que vive, das pessoas com quem ele se relaciona, das atividades que realiza em seu dia a dia, e com isso fazer suas escolhas.

A partir dos resultados gerados pelo consumidor ao concretizar a compra do produto, surge um terceiro momento relevante, que poderá retroalimentar o primeiro momento. Os resultados de produtos criados pelos consumidores no segundo momento, geram dados relacionados às preferências do público e do seu sistema cultural. Compreender melhor as informações desse consumidor podem contribuir com o primeiro momento do designer, de criação e na definição dos parâmetros iniciais, com dados que poderão retroalimentar seu processo criativo. Com o produto criado, o designer poderá utilizar as combinações propostas pelos consumidores para criar novas propostas, ou até mesmo disponibilizar aquelas que se tornaram mais relevantes dentro do processo de compra.

Figura 1 – Momentos da co-criação com retroalimentação de dados.



Fonte: A autora.

Os sistemas generativos permitem gerar complexidade, em que estão envolvidas diferentes ordens de grandeza, ou seja, a partir de elementos simples, podem ser criados outros elementos compostos. No atributo relacionado a cor, podem ser geradas inúmeras variações a partir dos parâmetros de matiz, saturação e luminosidade (HSB – *hue, saturation & brightness*), que definem a tonalidade, a quantidade de croma e de brilho de uma cor. E que por sua vez, podem ser combinadas em diferentes harmonias cromáticas, como as análogas, complementares, monocromática, entre outras, além de poderem ser definidos diferentes contrastes, como o alto, médio ou baixo. Ou seja, cada um desses parâmetros, ao serem combinados, geram outros parâmetros mais complexos, outras condições, criando componentes cada vez maiores. Segundo McCormarck et al. (2004) esses agregados podem gerar suas próprias interações, formando novos agregados com complexidade ainda maior, em uma hierarquia dinâmica.

A geração de complexidade é a primeira propriedade dos sistemas generativos, algo que pode ser percebido nas diferentes possibilidades de combinações de modelo, cores e estampas, em um sistema

A segunda propriedade é a relação complexa e interconectada entre organismos e ambiente, na qual a própria evolução dos organismos pode afetar e alterar o próprio ambiente. A complexa teia de relações entre as espécies, ou seja, entre os organismos, geram um ecossistema que possui ciclos de retroalimentação. Essa propriedade corrobora o terceiro momento proposto na figura 1 em que o designer se alimenta dos dados obtidos pelos resultados gerados pelos produtos, ao final da co-criação pelo consumidor.

A capacidade de automanutenção e auto reparação é a terceira dos sistemas generativos propriedade citada pelos autores (MCCORMARCK et al., 2004) e consiste na capacidade de adaptação em ambientes de mudança mantendo as configurações estáveis. A tolerância a falhas fornece a capacidade de superar mudanças que poderiam ser limitantes em outros tipos de projeto com características mais frágeis. Um exemplo disso pode ser justamente a alteração da cor do produto feita pelo consumidor, que em um sistema tradicional de varejo não poderia ser alterada, pois o produto já se encontraria produzido e disponível para a venda. Então, os “erros” de criação são solucionados antes mesmos de serem um problema. Outro exemplo, seria a mudança na largura do tecido, que poderá gerar problemas na fabricação de produtos em um sistema industrial tradicional, devido às configurações da estampa. Mas no contexto dos sistemas generativos, as padronagens geradas podem se adaptar facilmente à largura disponível, desde que considerada essa condição no desenvolvimento do programa.

A última capacidade dos sistemas generativos é o potencial em dar origem a propriedades genuinamente novas, comportamentos, resultados, estruturas ou relacionamentos. Designers estão sempre em busca de novidades, porém ao lidar com tendências de moda, por exemplo, pode cair em armadilhas ao ficar na “bolha” das tendências mundiais, sem trazer as especificidades do sistema cultural que o consumidor está inserido ou até mesmo componentes novos para a sua criação. Com esta capacidade, o design generativo abre as portas para resultados não previstos, o que pode transformar o varejo de moda a partir das interações com os consumidores, agregando mais diferenciações e revelando aspectos individuais das pessoas e que podem ser compartilhados por comunidades inteiras. Mas também pode ser um problema para designers que não estão acostumados com a imprevisibilidade dos seus projetos (MCCORMARCK et al., 2004).

Para lidar com a imprevisibilidade dos resultados, o designer pode usar como referência o próprio sistema evolutivo. Este sistema consiste em gerar uma gama de resultados iniciais a partir de um conjunto aleatório de parâmetros, suficiente para permitir visualizar uma ampla variedade de possíveis resultados. O senso estético do designer, baseado por conhecimentos técnicos da área de moda, conhecimentos sociais dos possíveis consumidores e referentes à análise da viabilidade produtiva, poderão auxiliar a determinar quais são os resultados “mais aptos” para continuarem no processo. Ao parametrizar aquilo que se destacou como “melhores resultados”, o novo conjunto de elementos irá herdar os traços dos seus “pais” bem-sucedidos.

Qualidades subjetivas como “beleza” ou “feiura” necessitarão sempre da interação humana, um gargalo para o designer neste contexto. Porém, se em um primeiro momento o designer se valer de conceitos pré-estabelecidos referentes à teoria de combinação de cores, por exemplo, o julgamento final será feito pelo consumidor, que irá retroalimentar o designer em próximos desenvolvimentos. Isso evidencia ainda mais a pertinência deste projeto no contexto da co-

criação, no qual a análise subjetiva fica a cargo do consumidor, que contribuirá com a bagagem do seu próprio sistema cultural.

As padronagens, exibidas em produtos de moda por meio de estamparia, em sua maioria, pode ser considerada a aplicação mais relevante do design generativo na moda, pois permite a visualização de formas, elementos e cores na composição visual. Sendo assim, este estudo irá focar a partir de agora no design de superfície e no desenvolvimento de padronagens para poder realizar reflexões e experimentações.

6 O design de superfície e o design generativo

O design de superfície é uma atividade criativa de projeto destinada a promover qualidades estéticas, funcionais e estruturais a uma superfície de um produto. Essas características devem estar adequadas ao processo produtivo e ao contexto sociocultural, e podem ser encontradas em diferentes produtos, desde papelaria, têxtil e até mesmo cerâmica, cada um com suas particularidades de design e produtivas. (Rüthschilling, 2008)

No design de superfície aplicada à moda, a padronagem são propagações de desenhos pela extensão de uma superfície, ou seja, desenhos propagados infinitamente pelo comprimento e largura de uma área. Na moda, a padronagem pode ser estampada ou tecida, ou seja, pode ser aplicada na superfície por meio da impressão de tinta ou por meio do entrelaçamento de fios previamente tingidos e que serão tecidos em uma ordenação específica para formar o desenho desejado, respectivamente.

A estamparia destaca-se entre as diferentes especialidades do design de superfície, pois a atuação do designer está atrelada à moda e à configuração da roupa. A estamparia consiste em um conjunto de processos de impressão que reproduzem o desenho para uma superfície têxtil, por meio de corantes, pigmentos, tintas e produtos químicos corrosivos e/ou isolantes (Laschuk e Rüthschilling, 2015, apud CARVALHO; RÜTHSCHILLING, 2016).

Os desenhos são elementos, ou motivos, que compõem as padronagens, e podem ser imagens, formas ou texturas, criados para formarem módulos, sendo este a menor unidade que inclui todos os elementos visuais que configuram a composição. A partir da criação dos módulos, utilizando conceitos como harmonia entre os motivos, prevendo pontos de encontro entre os módulos (contiguidade) e sua propagação (continuidade), são estabelecidos sistemas de repetição que irão gerar resultados para o padrão pretendido, para que tenha uma repetição visualmente contínua e única (Rüthschilling, 2008). No caso da estamparia, para que essa repetição aconteça em um sistema industrial de produção, é necessário um *rappor*, ou seja, um padrão de repetição que se propaga infinitamente no comprimento e na largura, formando encaixes perfeitos na junção de suas extremidades.

Porém, estampas sem *rappor* definido também podem ser produzidas desde que sejam projetadas de acordo com as dimensões da superfície a ser estampada. Ao considerar a área útil de um tecido necessária para a produção de um único produto, não há a obrigatoriedade de ter um *rappor*, mas uma área considerada no desenvolvimento do desenho a ser estampado. Neste caso, há um desenho projetado em função de uma área específica, não sendo considerado padronagem, já que não há a sua repetição. Este processo pode ser complexo ao considerar um processo de desenvolvimento de estampas tradicional, ao ter uma metragem grande de tecido para ser desenvolvida uma estampa que a preencha. Grandes arquivos terão que ser desenvolvidos e o designer terá que passar mais tempo projetando uma maior área para ser preenchida completamente por motivos. O design generativo pode solucionar este problema a partir da definição de parâmetros que irão “preencher” os espaços

pré-determinados com condições relacionadas à ocupação das formas e as cores determinadas. O próprio sistema poderá propagar um conjunto de elementos (módulo) dentro de um espaço, em um sistema de repetição pré-estabelecido, com condições que garantem sua propagação.

Com o uso de programação para o desenvolvimento de padronagens é possível usar o design generativo no design de superfícies sem que haja, necessariamente, um *rapport*, um padrão de repetição. O estudo de padronagens e estampas generativas já vem sendo realizado por diferentes profissionais do campo do design, promovendo reflexões acerca da possibilidade de gerar e produzir design de superfície com características generativas. O uso de linguagem de programação no design de superfície traz uma nova perspectiva ao designer que, ao invés de criar desenhos e apenas usar um programa para repeti-lo, ele irá informar ao programa os parâmetros que deseja para que atenda às características do projeto (LARANJEIRA et al., 2018).

Além disso, padronagens generativas podem representar a interação do consumidor com os parâmetros projetados pelo designer. A partir de parâmetros, o designer pode definir quais são as formas, as cores e o tamanho que determinado desenho formará a partir da interação com o usuário. Tamanho da tela, formato dos elementos, opções de cores são algumas características da padronagem que o programa computacional poderá proporcionar a partir das definições dos códigos projetados pelo designer.

Segundo Laranjeira et al. (2018) o design generativo e o pensamento computacional possibilitam explorar ideias e soluções inovadoras e, neste cenário, o design de superfície encontra ampla gama de possibilidades criativas. Possibilidades estas que permitem a criação de “novos métodos de criação de formas geométricas e/ou orgânicas” que promovem o “desenvolvimento de estampas com abordagens funcionais, estéticas e culturais diferenciadas”.

O design generativo está inserido dentro do design paramétrico, que por sua vez consiste em usar a programação de dados no desenvolvimento do projeto, com o “uso de parâmetros que carregam informações formais, estruturais, funcionais, estéticas e culturais para a concepção do design” (Yu; Gero; Gu, 2013 apud Laranjeira et al., 2018). Ou seja, os parâmetros são valores atribuídos na programação com o objetivo de informar características e propriedades ao objeto. Qualquer alteração de parâmetro pode alterar o resultado esperado. Além de simples códigos inseridos em um programa, o parametricismo é uma organização do pensamento que não pensa em uma simples solução para o problema, mas em estabelecer variáveis para que o computador se depare com inúmeros resultados possíveis.

Todo projeto de design é um projeto paramétrico, afirmam Laranjeira et al. (2018), pois todo projeto de design precisa de parâmetros para ser realizado. Diferente de sistemas CAD, nos quais reproduzem por meios digitais as técnicas analógicas de desenho, o entendimento de uma linguagem de programação permite o designer explorar diferentes e até mesmo desconhecidas possibilidades criativas, atreladas ao pensamento humano e à capacidade e eficácia do computador. Esse conhecimento pode ser aplicado em diferentes plataformas. Neste estudo, será utilizada a biblioteca p5.js, que permite gerar gráficos online a partir da linguagem de programação JavaScript.

Para o desenvolvimento de padronagens generativas é que o conjunto de códigos de linguagem de programação gerem um padrão. Para isso, é necessário considerar um conjunto de códigos como sendo o módulo da padronagem. Esse conjunto poderá se reproduzir

infinitamente, sem que haja necessariamente uma repetição, permitindo que o módulo se alterne completamente de maneira recorrente.

Na repetição do módulo, é importante que sejam consideradas algumas regras de movimentos, como rotação, translação e reflexão, e de sistemas de repetição, como bloco, tijolo e meio salto, que farão a propagação dentro de um conjunto de combinações específicas que não variam a partir de sua definição. O módulo pode ser criado a partir do código e a programação irá possibilitar uma evolução do módulo, fazendo com que cada módulo criado não seja igual ao anterior, podendo variar de tamanho e posição, mas ainda sim evoluindo e emergindo dentro da área estabelecida, conforme as variáveis e funções determinadas no sistema, sem necessariamente ter uma repetição.

Alterações de parâmetros poderão gerar mudanças nas características dos módulos em tempo real, como a cor e até mesmo a direção ou rotação dos elementos da padronagem. Essa característica dinâmica reforça os processos de design generativo como um método que valoriza a customização em massa (LARANJEIRA et al., 2018).

Casey Reas, um dos formuladores do software Processing desenvolveu trabalhos que são excelentes exemplos de como o design generativo pode ser aplicado no desenvolvimento de superfícies estampadas. A partir de um único código, podem ser apresentados diferentes resultados de acordo com os parâmetros inseridos. Mudanças na espessura das linhas, nas cores, tamanho, rotação e/ou adição de outros elementos, são possibilidades que surgem a partir do design generativo. Todas essas possibilidades geram oportunidades criativas para o designer de moda e a aplicação do design de superfície generativo em produtos do vestuário. As padronagens geradas deixam de ser estáticas e passam a ser dinâmicas, podendo ainda ter a participação do usuário na definição dos parâmetros.

Em um workshop virtual realizado por Barbara Castro, no canal Processing Brasil na plataforma do YouTube, é apresentada uma introdução à programação criativa utilizando a biblioteca p5.js, na plataforma OpenProcessing. Na transmissão, Castro (2021) demonstra a criação de padronagens gerados dinamicamente, inclusive com a participação de espectadores da ocasião. A criação dos padrões foi iniciada a partir das formas geométricas básicas e foram abordados conceitos como iteração, transformação e funções customizadas para a criação de módulos e das padronagens generativas. Ao final do workshop ela acrescentou códigos desenvolvidos por outros designers que estavam participando do workshop, mostrando que é possível criar uma interação com outras pessoas além do designer responsável pelo código principal. Antes de iniciar o programa, o que é visto são apenas letras e números, e assim que ele inicia, esses parâmetros se convertem em formas, cores e posições dentro da tela, de maneira estática ou animada. Os resultados passam a ser surpreendentes a partir do momento que o programa é iniciado.

6.1 Experimentações do design generativo no desenvolvimento de uma padronagem digital

A partir da contextualização teórica, será proposta neste estudo uma experimentação prática de programação de uma padronagem generativa, utilizando o programa p5.js, uma plataforma online que possui sua própria biblioteca de referência para os códigos utilizados. O desenvolvimento se iniciou a partir da escolha de um tema para a criação dos primeiros parâmetros relacionados à forma e cores. A partir de pesquisas realizadas na plataforma de pesquisa de tendências WGSN e da observação de perfis do Instagram de influenciadores digitais, foi escolhido o tema de “otimismo”.

O tema otimismo trata da busca de momentos que possam trazer melhores expectativas e até mesmo gerar sensações de alegria após um longo período de isolamento social e incertezas. Formas geométricas irregulares, movimentos orgânicos, cores saturadas e a transparência promovem elementos que trazem sensações de liberdade e positividade.

A definição das cores foi feita usando o sistema HSB (Hue, Saturation and Brightness) em uma escala de 0 a 100. Foi feito um estudo preliminar de quais códigos correspondem às cores finais mostradas pelo programa. Para facilitar a parametrização, foi elaborada uma tabela com duas opções de cores: a primeira considerando valores máximos de saturação e brilho e a segunda, considerando valores baixos de saturação para obtenção de cores “pastéis”.

Figura 2 – Tabelas de referência para códigos de cor no sistema HSB.

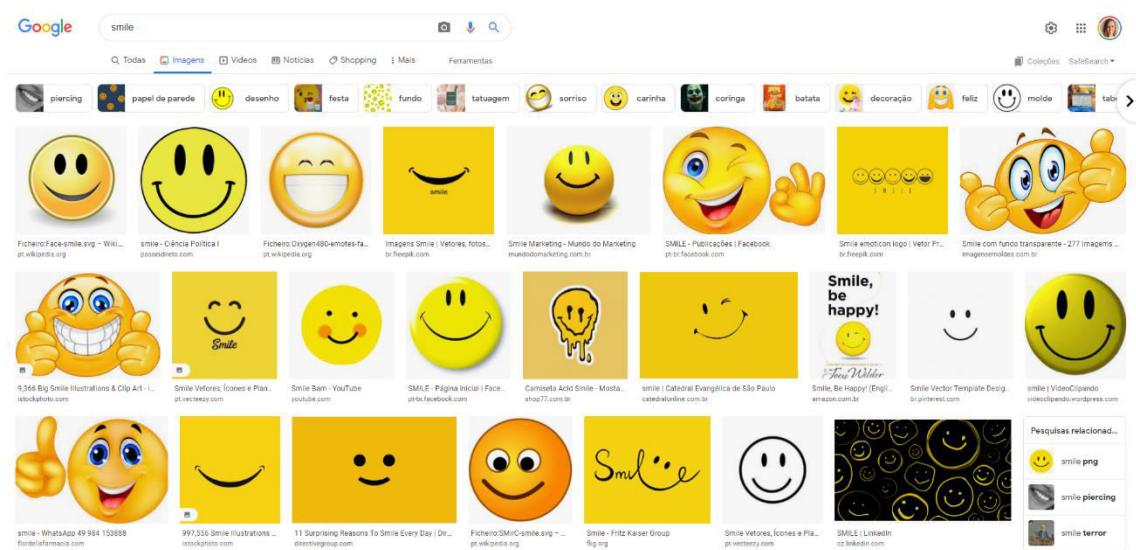
color											
H	0	10	20	30	40	50	60	70	80	90	100
S	100	100	100	100	100	100	100	100	100	100	100
B	100	100	100	100	100	100	100	100	100	100	100

color											
H	0	10	20	30	40	50	60	70	80	90	100
S	30	30	30	30	30	30	30	30	30	30	30
B	100	100	100	100	100	100	100	100	100	100	100

Fonte: A autora.

Para a definição das formas foram escolhidas, inicialmente, aquelas que possibilitam desenhar o formato de um “smile” (sorriso), um círculo com dois olhos e uma boca, símbolo que representa felicidade na cultura pop e utilizado em sistemas de mensagem para expressar emoções. Em uma pesquisa direta no google, foram obtidas algumas imagens para referência de forma e cor.

Figura 3 – Resultados da pesquisa no google pelo termo “smile”.



Fonte: Google.

Foi desenvolvido um código que apresentasse um *smile* ainda sob efeitos da pandemia, não simétrico e com formas ligeiramente diferentes umas das outras, como se estivesse ainda se recuperando dos momentos difíceis e confusos. O corpo do *smile* foi criado sem o contorno, para torná-lo mais suave. Os olhos foram criados diferentes um dos outros, para evidenciar a recuperação da felicidade e para explorar mais possibilidades criativas na criação dos códigos.

Figura 4 – Códigos para o desenho do smilouco.

```

1  function setup() {
2    createCanvas(200, 200);
3    background(240);
4    colorMode(HSB,100);
5  }
6
7  function draw() {
8    //corpinho
9    fill(14,100,100);
10   noStroke();
11   circle(100,100,120);
12
13  //olho esquerdo, parte branca
14  fill(100);
15  stroke(0);
16  strokeWeight(1);
17  ellipse(70,100,30,35);
18
19  //olho direito, parte branca
20  fill(100);
21  stroke(0);
22  strokeWeight(1);
23  ellipse(120,80,30,25);
24
25  //olho esquerdo, parte preta
26  fill(0);
27  stroke(0);
28  strokeWeight(1);
29  ellipse(72,98,16,18);
30
31  //olho direito, parte preta
32  fill(0);
33  stroke(0);
34  strokeWeight(1);
35  ellipse(120,77,15,10);
36
37  //boca
38  noFill();
39  strokeWeight(4);
40  arc(110, 115, 50, 50, 0, HALF_PI);
41 }
```



Fonte: A autora in p5.js (2022). Disponível em: <https://editor.p5js.org/divalucosta/sketches/gvgHnXEPT>

Foi criada uma segunda opção para o *smile* usando o mesmo código, apenas trocando a função da forma de elipse por retângulo, pois ambas possuem o mesmo número de parâmetros. Porém, no caso do retângulo, é necessário especificar o modo de desenho centralizado, pois o programa desenha retângulos a partir do canto esquerdo superior. Abaixo podem ser visualizadas as duas opções de *smile* com formas retangulares, a primeira sendo o modo automático do retângulo, e a segunda, com o código para desenho centralizado.

Figura 5 – Códigos para o desenho do smilouco com formas retangulares.

```

1 function setup() {
2   createCanvas(200, 200);
3   background(240);
4   colorMode(HSB,100);
5 }
6
7 function draw() {
8   //corpinho
9   fill(14,100,100);
10  noStroke();
11  rect(100,100,120);
12
13  //olho esquerdo, parte branca
14  fill(100);
15  stroke(0);
16  strokeWeight(1);
17  rect(70,100,30,35);
18
19  //olho direito, parte branca
20  fill(100);
21  stroke(0);
22  strokeWeight(1);
23  rect(120,80,30,25);
24
25  //olho esquerdo, parte preta
26  fill(0);
27  stroke(0);
28  strokeWeight(1);
29  rect(72,98,16,18);
30
31  //olho direito, parte preta
32  fill(0);
33  stroke(0);
34  strokeWeight(1);
35  rect(120,77,15,10);
36
37  //boca
38  noFill();
39  strokeWeight(4);
40  arc(110, 115, 50, 50, 0, HALF_PI);
41 }

```

```

1 function setup() {
2   createCanvas(200, 200);
3   background(240);
4   colorMode(HSB,100);
5   rectMode(CENTER);
6 }
7
8 function draw() {
9   //corpinho
10  fill(14,100,100);
11  noStroke();
12  rect(100,100,120);
13
14  //olho esquerdo, parte branca
15  fill(100);
16  stroke(0);
17  strokeWeight(1);
18  rect(70,100,30,35);
19
20  //olho direito, parte branca
21  fill(100);
22  stroke(0);
23  strokeWeight(1);
24  rect(120,80,30,25);
25
26  //olho esquerdo, parte preta
27  fill(0);
28  stroke(0);
29  strokeWeight(1);
30  rect(72,98,16,18);
31
32  //olho direito, parte preta
33  fill(0);
34  stroke(0);
35  strokeWeight(1);
36  rect(120,77,15,10);
37
38  //boca
39  noFill();
40  strokeWeight(4);
41  arc(110, 115, 50, 50, 0, HALF_PI);

```

Fonte: A autora in p5.js (2022). Disponível em: <https://editor.p5js.org/divalucosta/sketches/v-DONUovk>

Baseado no efeito gerado pela “não centralização” das formas, foi feito um terceiro teste com a elipse alinhada pelo canto. O resultado pode ser visualizado abaixo. Vale ressaltar que, para esse efeito, é necessário incluir o código para este tipo de modo de criação da elipse.

Figura 6 – Código para o desenho do smilouco com elipses alinhadas pelo canto.

```

1 function setup() {
2   createCanvas(300, 300);
3   background(240);
4   colorMode(HSB);
5   ellipseMode(CORNER);
6 }
7
8 function draw() {
9   //corpinho
10  fill(50,100,100);
11  noStroke();
12  circle(100,100,120);
13
14  //olho esquerdo, parte branca
15  fill(100);
16  stroke(0);
17  strokeWeight(1);
18  ellipse(70,100,30,35);
19
20  //olho direito, parte branca
21  fill(100);
22  stroke(0);
23  strokeWeight(1);
24  ellipse(120,80,30,25);
25
26  //olho esquerdo, parte preta
27  fill(0);
28  stroke(0);
29  strokeWeight(1);
30  ellipse(72,98,16,18);
31
32  //olho direito, parte preta
33  fill(0);
34  stroke(0);
35  strokeWeight(1);
36  ellipse(120,77,15,10);
37
38  //boca
39  noFill();
40  strokeWeight(4);
41  arc(110, 115, 50, 50, 0, HALF_PI);

```

Fonte: A autora in p5.js (2022).

Com as possíveis formas desenvolvidas, a próxima etapa é organizar o código em um módulo. Para isso, foi criada a função “modulo”. Todo o código foi inserido nessa nova função, nomeada “function modulo”. E para que ela seja desenhada novamente, é necessário inserir na função de desenho (function draw), o comando para que o módulo seja desenhado: modulo ();.

Figura 7 – Criação de um módulo.



```

1  function setup() {
2    createCanvas(1000, 1000);
3    background(240);
4    colorMode(HSB);
5    //ellipseMode(CORNER);
6  }
7
8  function modulo() {
9    //corpinho
10   fill(50,100,100);
11   noStroke();
12   circle(100,100,120);
13
14   //olho esquerdo, parte branca
15   fill(100);
16   stroke(0);
17   strokeWeight(1);
18   ellipse(70,100,30,35);
19
20   //olho direito, parte branca
21   fill(100);
22   stroke(0);
23   strokeWeight(1);
24   ellipse(120,80,30,25);
25
26   //olho esquerdo, parte preta
27   fill(0);
28   stroke(0);
29   strokeWeight(1);
30   ellipse(72,98,16,18);
31
32   //olho direito, parte preta
33   fill(0);
34   stroke(0);
35   strokeWeight(1);
36   ellipse(120,77,15,10);
37
38   //boca
39   noFill();
40   strokeWeight(4);
41   arc(110, 115, 50, 50, 0, HALF_PI);
42 }
43
44 function draw() {
45   modulo ();
46 }
```

Fonte: A autora in p5.js (2022). Disponível em: <https://editor.p5js.org/divalucosta/sketches/gygHnXEPT>

Para fazer a repetição do módulo é necessário inserir parâmetros que irão fazer a translação do módulo pela largura e altura da tela. Para isso, é necessário inserir um conjunto de parâmetros, começando com uma instrução *for*. De acordo com a biblioteca de referências do p5.js, o “*for*” consiste em executar uma seção de códigos várias vezes, criando um “loop” (laço de repetição). Ele possui três expressões diferentes dentro do parêntesis, sendo a primeira para definir o estado inicial do loop, a segunda para expressar uma condição que deve ser verificada antes de cada loop, e a terceira, é a expressão para executar o loop, quando a segunda expressão retornar verdadeira. No caso da segunda expressão retornar falsa, o loop é encerrado, ou seja, não é executado. É importante criar uma condição que eventualmente seja considerada como “falsa” para que o programa possa interromper o fluxo, senão, ele irá repetir infinitamente e poderá “travar” o computador.

No caso da padronagem, foi criada uma condição de repetição de 4 vezes, para que o loop repita apenas 4 vezes o *smile*, tanto no eixo de x, quanto no eixo de y. E para que os módulos

não fiquem sobrepostos, a translação foi programada para uma distância de 130, ou seja, 10 a mais do que o tamanho do círculo externo do *smile*.

Figura 8 – Repetição do módulo.

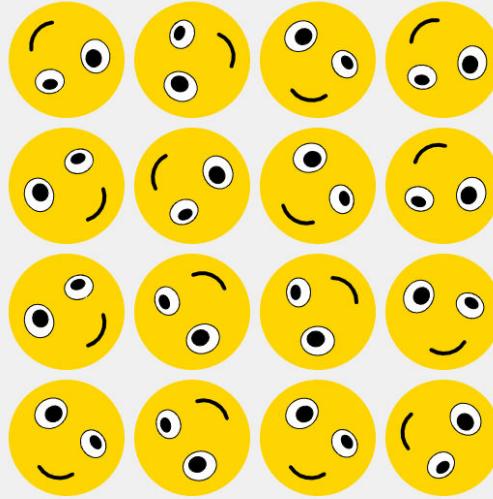


Fonte: A autora in p5.js (2022). Disponível em: <https://editor.p5js.org/divalucosta/sketches/n6c1bYwou>

Para criar um aspecto mais aleatório na padronagem, foram inseridos códigos de rotação utilizando a criação de uma variável. A variável é iniciada por “*let*” seguida de um nome criado pelo designer. Neste caso, foi criada uma variável “*rot*”, para referir-se à “rotação”. Primeiramente a variável é declarada, para depois ser utilizada no código. Para definir os parâmetros da rotação, foram usados os códigos de translação a partir da mesma posição inicial do círculo (100, 100).

Ao final do código, para que não fique rotacionando infinitamente, foi inserido o “*noLoop*”, para que, cada vez que o programa for iniciado, uma nova rotação seja feita em todos os *smiles*. Caso contrário, os *smiles* ficariam rotacionando 60 vezes por segundo.

Figura 9 – Rotação dos módulos na repetição.



Fonte: A autora in p5.js. Disponível em: <https://editor.p5js.org/divalucosta/sketches/6IQcaRSKn>

Além disso, é possível tornar aleatória a repetição do módulo, utilizando o parâmetro “random”. Para isso, foi aumentada a distância entre os módulos e em seguida, somado o *random*, com um valor que não sobreponha os módulos. Foi utilizado o valor de 30, correspondente a diferença entre a posição (100) e a translação (130).

Nesse caso, a repetição dos módulos não segue a simetria da translação, e sim uma repetição aleatória, assimétrica, com espaçamentos diferentes entre os módulos. Toda vez que o programa é iniciado, uma nova configuração aleatória de translação é apresentada. Essa característica da padronagem generativa é unicamente derivada da programação, não sendo possível fazer algo aleatoriamente infinito em um desenvolvimento de padronagem “tradicional”, utilizando um *rapport*.

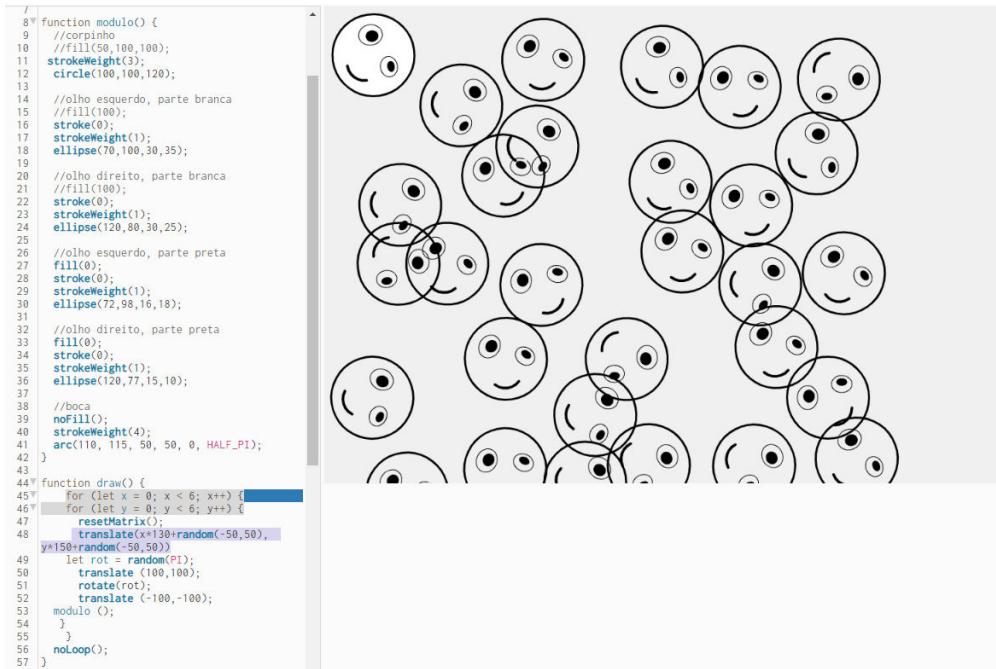
Figura 10 – Repetição aleatória.



Fonte: A autora in p5.js. Disponível em: <https://editor.p5js.org/divalucosta/sketches/2Q2eRVBv5>

Dante da possibilidade de tornar aleatória a translação, também pode ser definido um intervalo no qual a característica aleatória irá se manter. No exemplo anterior, mantendo a característica de não sobrepor nenhum módulo no outro, poderia ser definido como *random* (-15, 15). Caso a sobreposição não seja um problema, podem ser programados outros resultados, que poderão se tornar interessantes dependendo da configuração dos motivos no módulo. Para uma visualização diferente, no exemplo abaixo, foram descartadas as informações de preenchimento em cores, mantendo apenas o preenchimento preto dos olhos e o contorno das formas. Além disso, foi aumentado o número de repetições do módulo e aumentada a abrangência do *random*, para permitir a sobreposição de módulos. O mesmo código foi utilizado, apenas descartando as informações de preenchimento e alterando os parâmetros do *random*, como pode ser visualizado a seguir.

Figura 11 – Repetição aleatória dos módulos transparentes e com sobreposição.



Fonte: A autora in p5.js. Disponível em: <https://editor.p5js.org/divalucosta/sketches/3kiBuMceX>

A característica aleatória também pode ser usada para outros parâmetros, como a cor. A partir do estudo de cores realizado no início desta seção, é possível definir variáveis para a cor. Com o uso de “*let*”, é possível definir valores específicos para H, S e B no início do código. No exemplo a seguir, o valor de H, responsável pela matiz, foi definido como aleatório dentro de uma escala de 0 a 100. Para que essa escala seja considerada, é necessário declarar na função de setup, que o sistema HSB corresponde a uma escala de 100. Caso contrário, ele irá considerar a escala de 360 cores. Já a variável de S, foi considerada com valor de 30, para que

as cores ficassem um pouco menos saturadas e com características de cores “pastéis”. E a variável B foi considerada como 100, para que as cores ficassem claras.

Figura 12 – Padronagem com cores aleatórias.

```
function setup() {
  createCanvas(1000, 1000);
  background(240);
  colorMode(HSB,100);
}

function module() {
  //corpinho
  let h,s,b;
  h = random(100);
  s = 30;
  b = 100
  fill(h,s,b);
  noStroke();
  circle(100,100,120);

  //olho esquerdo, parte branca
  fill(100);
  stroke(0);
  strokeWeight(1);
  ellipse(70,100,30,35);

  //olho direito, parte branca
  fill(100);
  stroke(0);
  strokeWeight(1);
  ellipse(120,80,30,25);

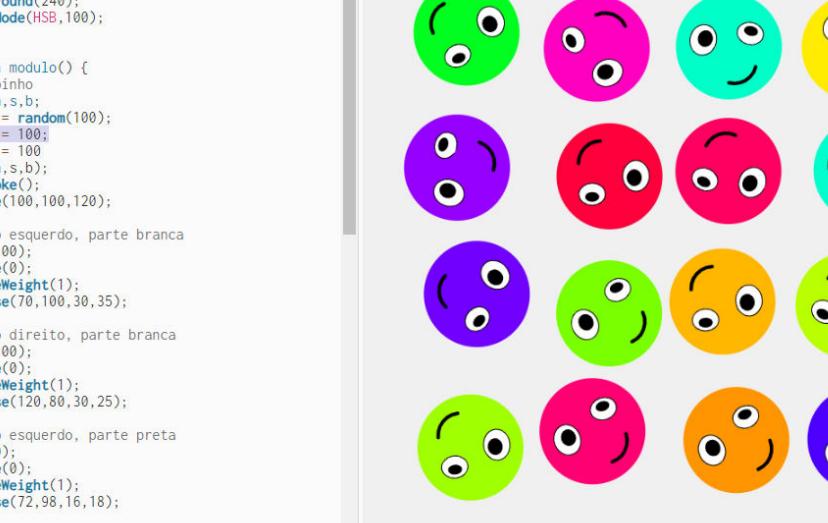
  //olho esquerdo, parte preta
  fill(0);
  stroke(0);
  strokeWeight(1);
  ellipse(72,98,16,18);

  //olho direito, parte preta
  fill(0);
  stroke(0);
  strokeWeight(1);
  ellipse(120,77,15,10);
}
```

Fonte: A autora in p5.js. Disponível em: <https://editor.p5js.org/divalucosta/sketches/Wib8DnamF>

Foi realizado um outro experimento com as cores totalmente saturadas, e ainda assim, aleatórias. O resultado é apresentado a seguir.

Figura 13 – Padronagem com cores aleatórias saturadas.



The image shows a 4x4 grid of 16 smiley faces, each consisting of a colored circle with two black eyes and a curved line for a mouth. The colors of the circles are arranged in a repeating pattern: green, pink, cyan, yellow, purple, red, magenta, cyan, purple, green, orange, lime green, blue, red, orange, lime green, and blue.

```
1 function setup() {
2   createCanvas(1000, 1000);
3   background(240);
4   colorMode(HSB,100);
5 }
6
7 function modulo() {
8   //corpinho
9   let h,s,b;
10  h = random(100);
11  s = 100;
12  b = 100
13  fill(h,s,b);
14  noStroke();
15  circle(100,100,120);
16
17  //olho esquerdo, parte branca
18  fill(100);
19  stroke(0);
20  strokeWeight(1);
21  ellipse(70,100,30,35);
22
23  //olho direito, parte branca
24  fill(100);
25  stroke(0);
26  strokeWeight(1);
27  ellipse(120,80,30,25);
28
29  //olho esquerdo, parte preta
30  fill(0);
31  stroke(0);
32  strokeWeight(1);
33  ellipse(72,98,16,18);
34
35  //olho direito, parte preta
36  fill(0);
37  stroke(0);
38  strokeWeight(1);
39  ellipse(120,77,15,19);
```

Fonte: A autora in p5.js. Disponível em: <https://editor.p5js.org/divalucosta/sketches/BX-vpNPM3>

A partir destes experimentos é possível verificar algumas possibilidades de parametrizar uma série de informações que permitirão a interação com o consumidor, seja pelas cores, pelas formas ou até mesmo pela própria maneira de repetição dos módulos. Assim, nenhuma

padronagem será igual a outra, ou seja, a customização em massa se torna realidade a partir do design generativo e que poderá ter a interação de consumidores, utilizando uma interface que permita fazer as alterações de maneira mais intuitiva.

Não foi mencionado neste estudo, mas ainda existem possibilidades de interação a partir do movimento do mouse, de cliques com o mouse e com o pressionar de teclas do teclado. São possibilidades que ampliam os resultados para padronagens geradas por design generativo.

7 Considerações Finais

Neste cenário, o papel do designer vai além da visão tradicional do design, em que a relação entre o profissional e o artefato é direta, ou seja, existe uma relação entre as intenções do designer e a do artefato produzido. No design generativo, as regras e sistemas envolvidos são os grandes propulsores de interação para gerar o design acabado. O designer manipula os parâmetros para gerar o artefato e não o artefato em si, no qual o artefato é resultado das propriedades emergentes do processo de interação entre os parâmetros, e no caso deste estudo, de pessoas não designers, ou seja, os próprios usuários do produto.

Explorar e organizar as possibilidades do design generativo neste estudo, permite vislumbrar uma possível aplicação no contexto da indústria 4.0, por meio de um projeto de auto-organização e automontagem. De acordo com McCormack, Dorin & Innocent (2004), os sistemas de automontagem consistem em modelar ou construir um grande número de elementos simples que se combinam para construir artefatos de larga escala, em várias configurações.

As experimentações realizadas a partir da reflexão teórica de conceitos aplicados ao design generativo, ao design de superfície, ao design de moda e à indústria 4.0, permitem divulgar e ampliar o alcance da integração entre essas áreas, possibilitando o surgimento de novos projetos de design generativo na moda.

Por não ter um método formalizado ou instrucional, o design generativo pode ser nomeado também como “arte generativa”. Porém, como o papel do designer permanece e se faz relevante para dominar a relação entre a especificação do processo, o ambiente e o artefato gerado, o design generativo pode ser considerado uma evolução da profissão já conhecida do designer. Apesar de ser uma evolução, o designer deve considerar a coerência do projeto perante as transformações físicas ou comportamentais. Ou seja, no contexto desta pesquisa, considerar a própria evolução da indústria e as mudanças no sistema cultural do consumidor, para que possa manter seu caráter inovador e relevante em relação à experiência proporcionada ao usuário.

8 Referências

ADIDAS. **Knit for you.** Disponível em: <http://adidasknitforyou.com/#experience> Acesso em: 27 fev. 2022.

BRUNO, Flávio da Silveira. **A quarta revolução industrial do setor têxtil e de confecção/ a visão de futuro para 2030.** 2ª Ed. São Paulo: Estação das Letras, 2017.

CARVALHO, Nathalia Alborghetti; RÜTHSCHILLING, Evelise Anicet. Inovação em metodologia de projeto aplicada ao design de superfície voltado para moda. **ModaPalavra e-periódico**, Volume 9, n.17, p. 179-194, jan-jun, 2016. Disponível em:
<https://www.revistas.udesc.br/index.php/modapalavra/article/view/1982615x09172016179>
Acesso em: 19 fev. 2022

CASTRO, Bárbara. **Workshop de padronagens generativas.** YouTube, Processing Community Day Brasil, 12 abr. 2021. Disponível em: <https://www.youtube.com/watch?v=e2GCaV6ERKI> Acesso em: 19 fev. 2022.

FRALIX, Michael T. From mass production to mass customization. **Journal of Textile and Apparel, Technology and Management**, NC State University, Volume 1, issue 2, Winter 2001. Disponível em: https://textiles.ncsu.edu/tatm/wp-content/uploads/sites/4/2017/11/fralix_full.pdf Acesso em: 19 fev. 2022.

LARANJEIRA, Mariana Araujo; MARAR, João Fernando; PASCHOARELLI, Luis Carlos; LANDIM, Paula da Cruz. Design Generativo de Superfícies: uma análise do uso de programação para o desenvolvimento de estamparia. **ModaPalavra e-periódico**, Volume 11, n.21, p. 005-020, jan-jun, 2018. Disponível em: <https://www.revistas.udesc.br/index.php/modapalavra/article/view/10371> Acesso em: 19 fev. 2022

MALIK, Aamina Karim. **Mass Customization!** An approach through Generative Design. In: Generative Art Conference Proceedings, Milan, 2016.

McCORMACK, Jon; DORIN, Alan; INNOCENT, Troy. **Generative design: a paradigm for design research.** In: Redmond, J. et. al. (eds) Proceedings of Futureground, Design Research Society, Melbourne, 2004.

NIKE. Nike by you. Disponível em: <https://www.nike.com/nike-by-you> Acesso em: 27 fev. 2022.

P5.JS. Reference. Disponível em: <https://p5js.org/reference> Acesso em: 27 fev. 2022.

PRIBERAM. **Síntese.** Disponível em: <https://dicionario.priberam.org/s%C3%ADntese> Acesso em: 19 fev. 2022.

RUTHSCHILLING, Evelise Anicet. **Design de superfície.** Porto Alegre: UFRGS, 2008.