

PUC Rio, Pontifícia universidade católica do  
Rio de Janeiro / DSG1042

Aluno: Antonio Carlos Santos Thiele  
Orientador e Prof.Pauta: João Bonelli

# Agradecimentos

Ao meu orientador João Bonelli, por acreditar e apoiar esta proposta de projeto mesmo antes do começo da matéria.

Ao professor Jorge lopes, por seus sábios conselhos.

Ao professor Leonardo Cardarelli pelas tutorias e métodos para organizar os trabalhos.

À Professora Claudia Bolshaw pelas dinâmicas e conselhos.

Aos demais professores da matéria por todo o suporte e atenção.

À Márcia Couto e Raquel Mota, pelo grande apoio em todos os momentos e pela ajuda com revisão de textos e organização das ideias.

À monitora Antonia Muniz pelo suporte dentro e fora da sala de aula.

# Sumário

1.Introdução -----	P8
1.1 Proposta e objetivos-----	P6
1.2 Relevância e justificativa-----	P9
1.3 Público alvo-----	P9
1.4 Metodologia-----	P10
2.Análises e pesquisas-----	P11
2.1 levantamento de dados-----	P12
2.2 Pesquisa teórica-----	P13
2.3 Pesquisa de similares -----	P17
2.4 Pesquisa estética-----	P25
2.5 Pesquisa técnica e tecnológica-----	P30
3.Experimentos cíclicos-----	P40
3.1 Primeiro experimento, controlador -----	P41
3.2 Segundo experimento, fita-----	P44
3.3 Terceiro experimento Modelagem com primitivas -----	P46
3.4 Quarto experimento Modelagem poligonal triangular -----	P53
3.5 Quinto experimento, Escultor -----	P56
3.6 Escolha de um caminho e Passos futuros -----	P58
4. testes com usuários -----	p60
4.1 Primeiro teste -----	p60
4.2 Segundo teste -----	p62
4.3 Terceiro teste -----	p67
4.4 Quarto teste -----	p69
5.Instalação -----	p71
5.1 Desenho da instalação -----	p72
5.2 Montagem -----	p74
6.Considerações gerais e passos futuros -----	p77
7.Referências Bibliográficas -----	p78

# Lista de figuras

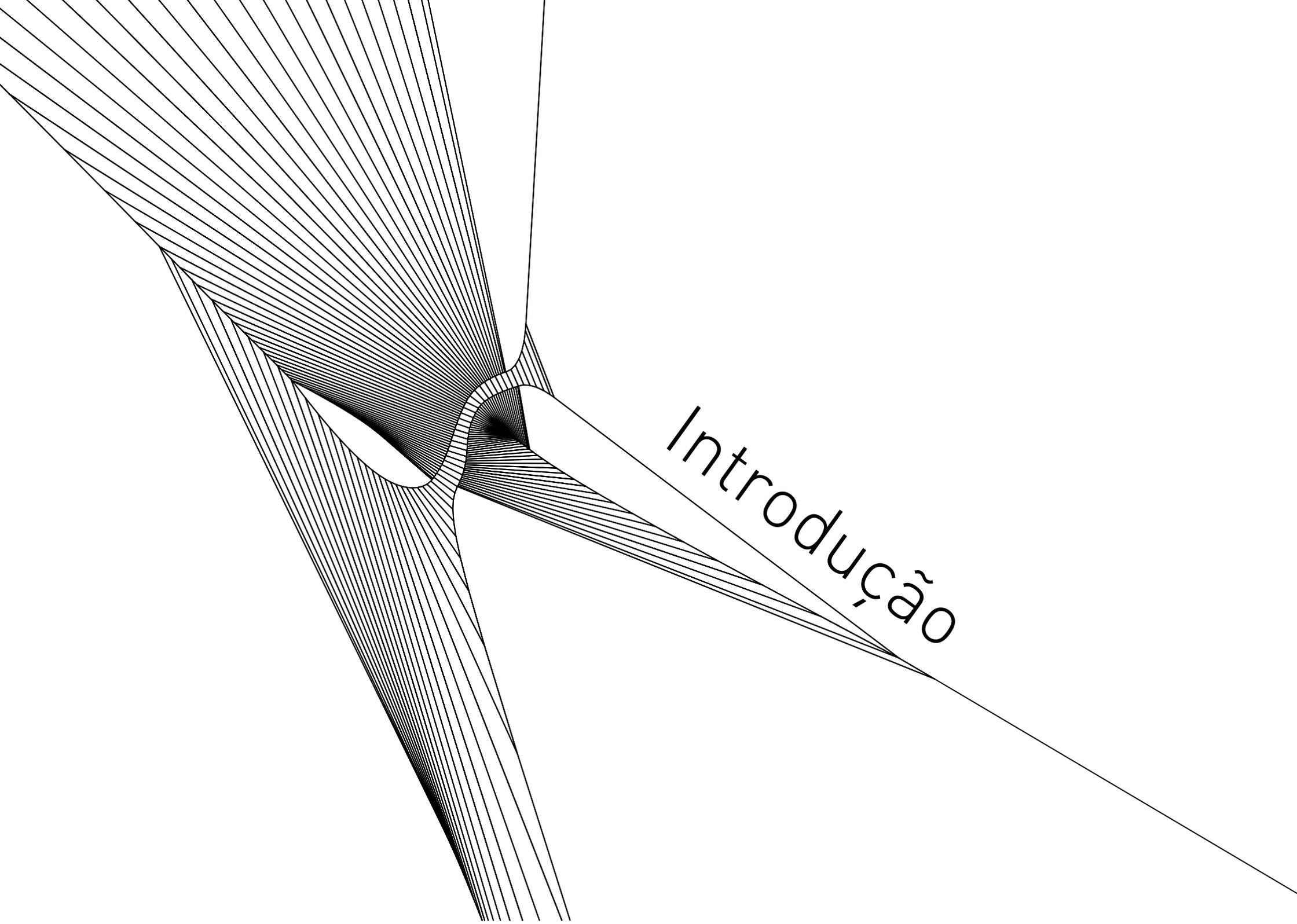
1. Obra de abertura do vídeo How To Draw With Code, 2012 -----	p14
2. Art basel Miami, With Barry Friedman Gallery LTD, 2005 -----	p17
3. Objetos prototipados com uso de laser -----	p17
4. Artisan electronique, instalação realizado em parceria com Tim Knapen e RepRap community, 2010 -----	p18
5. 3D Tracking Interface,Nick Normal, 2010 -----	p19
6. software CRE8 em uso -----	p20
7. Novint falcon com a pistola acoplada -----	p20
8. projeto do MIT Media Lab T(ether) -----	p21
9. Mutator, William latham, 1985 -----	p22
10. Espetáculo de dança Seventh Sense (Excerpt) -----	p23
11. Usuário modelando com "Kit move" -----	p24
12."Kit move" -----	p24
13. Artisan electronique, instalação realizado em parceria com Tim Knapen e RepRap community, 2010 -----	p25
14. Process 10 2005, Software, variable Size -----	p26
15. Modelo gerado parametricamente -----	p35
16. Número mínimo de iterações -----	p35
17. Voronoi em suas formas -----	p39
18. Arduino e sensores -----	p42
19. Número mínimo de iterações -----	p42
20. Número médio de iterações -----	p43
21. Número máximo de iterações -----	p43
22. Fita de malha mesh 3D -----	p44
23. Relação do usuario com o modelo -----	p44
24. cubos de tamanho fixo -----	p47
25. cubos com escala e rotação -----	p47
26. Montagem do test -----	p48
27. desenho cego Juliana Machado -----	p48

# Lista de figuras

28. Juliana Machado, modelando -----	p49
29. cubos cobrindo o modelador -----	p49
30. marca de distância mínima -----	p50
31. área de captura -----	p50
32. Efeito compasso no eixo Z -----	p51
33. Efeito compasso no eixo Y -----	p51
34. desenho cego Marcello Cortez -----	p52
35. desenho vendo Marcello Cortez -----	p52
36. Primeiro triangulo -----	p53
37. fita de malha triangular -----	p53
38. Malha com buraco representação 2D -----	p54
39. Reparando a malha com "snap" -----	p54
40. Objeto fechado -----	p55
41. cursor 3D -----	p56
42. Modelo gerados -----	p56
43. Modelo virtual -----	p57
44. Modelo prototipado -----	p57
45. Relação do usuario com o modelo -----	p58
46. posição ideal para o cubo ser calibrado -----	p63
47. Aluno Thed Oliveira testando os limites do "clap" -----	p63

# Lista de figuras

48. Primeiro menu -----	p65
49. teste com Rafael Crespo -----	p65
50. Modelagem em Wireframe -----	p66
51. Borracha em uso -----	p66
52. representação das mãos -----	p67
53. Possibilidades de mira -----	p67
54. Menu modificado -----	p68
55. Aluno Vítor Pimentel modelando com a borracha -----	p68
56. menu final -----	p70
57. Opções de ícones -----	p70
58. estrutura montada -----	p74
59. peças de truss -----	p74
60. Projetor fixo no truss -----	p76



Introdução

# 1. Introdução

Este trabalho tem como objetivo explorar a interação humano-computador com o fim de gerar e manipular modelos tridimensionais, apoiando-se no conceito de design de interação, para promover inovações tanto no âmbito do hardware quanto do software.

Acreditamos que, a partir da experimentação prática, surgirão formas originais de modelar que futuramente poderão ser aplicadas como novas técnicas pela indústria, por designers, por artistas ou por qualquer pessoa que trabalhe com modelagem tridimensional.

A indústria, nesse caso composta por fabricantes de programas e equipamentos específicos para modelagem, é inovadora até certo ponto. Ela não pode lançar a todo o momento produtos completamente diferentes para a mesma função pois deve arcar com limitações; como custo de produção e a aceitação dos mesmos pelo mercado. Isso trava o desenvolvimento de alternativas para os atuais paradigmas. Porque muitas vezes, essas companhias optam por evoluir seus produtos linearmente, onde os lançamentos

mais modernos são essencialmente os anteriores com algumas modificações pontuais. Observa-se que o ambiente acadêmico é ideal como alternativa a esse modelo. Uma vez que, estando livre das obrigações da indústria, podemos experimentar alternativas totalmente diferentes das anteriores.

Devido a isso cabe a pergunta: Como podemos criar, no ambiente acadêmico, ferramentas de interação específicas para modelagem virtual que nos possibilitem uma melhor compreensão e uso do espaço e volume tridimensional virtual?

## 1.1 Proposta e objetivos

Através de um projeto de design de interação, que busca explorar diferentes maneiras de comunicação entre o homem e o computador. Estruturado por uma metodologia de experimentos cíclicos. Onde são feitas propostas de novas formas de interação com o ambiente virtual, que posteriormente são desenvolvidas por meio de experimentos práticos. Com o intuito desenvolver uma ferramenta de modelagem que possibilite uma melhor compreensão de uso do espaço e volume tridimensional virtual.

## 1.2 Relevância e justificativa

Atualmente as interfaces de programas para modelagem utilizam os “inputs” físicos de maneira convencional, onde um teclado é usado para digitar comandos e formas; enquanto mouse ou caneta digitalizadora expressam contornos que atuam na modelagem.

Softwares como Z Brush e 123D Sculp apresentam inovações com relação à entrada de dados se comparados a programas tradicionais, pois modelam a partir de caneta digitalizadora e de “tablets”, respectivamente. Porém existem vários tipos de tecnologias baseadas em sensores e em outros recursos de interação que poderiam ser explorados na construção de tais interfaces.

Os exemplos atuais não levam em conta o corpo humano como um todo em suas possibilidades de comunicação. Ferramentas de vanguarda, como é o caso do controle de videogame ( para o console Xbox 360 ) Kinect, que é basicamente uma câmera tridimensional, e sensores como acelerômetros, podem ser usados para captar os movimentos humanos e convertê-los em comandos para manipular malhas virtuais.

## 1.3 Público alvo

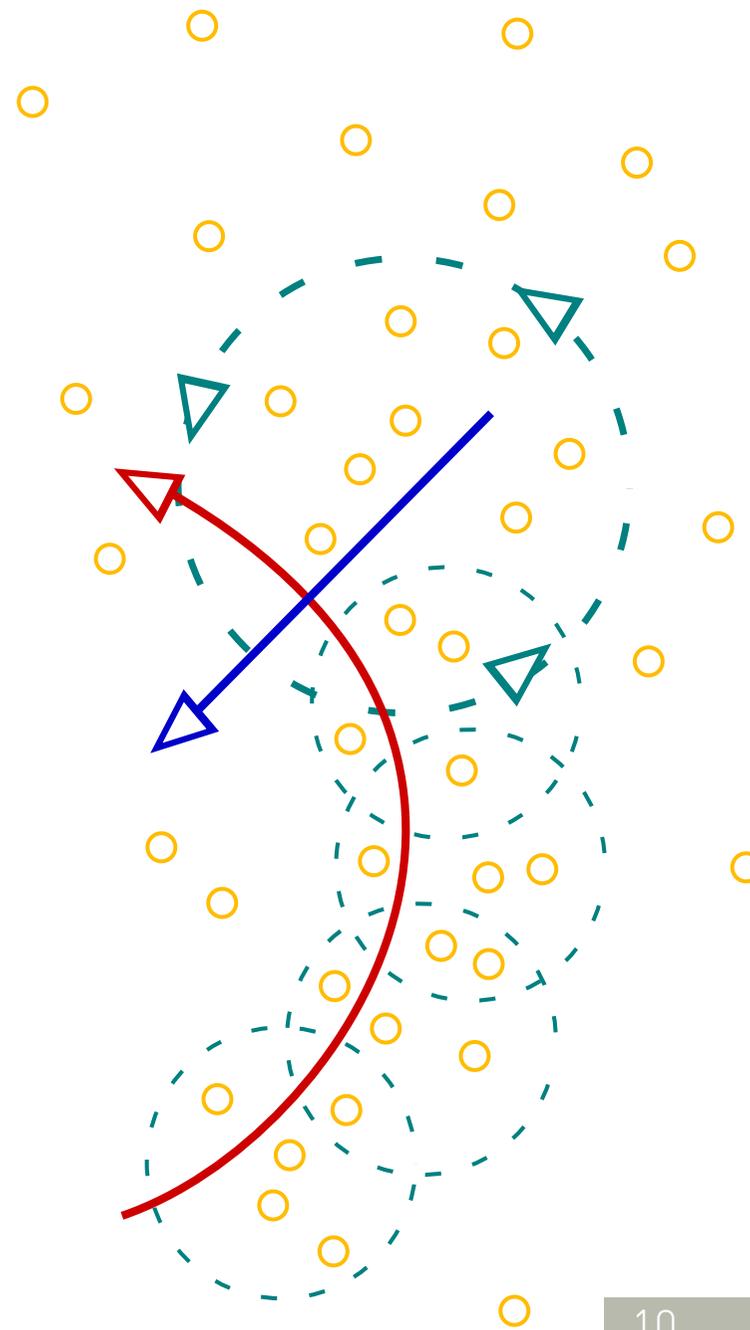
Todos os usuários, com limitação física ou não, que lidam com a transposição do meio de trabalho bidimensional para o universo tridimensional, com o intuito de manipular modelos virtuais.

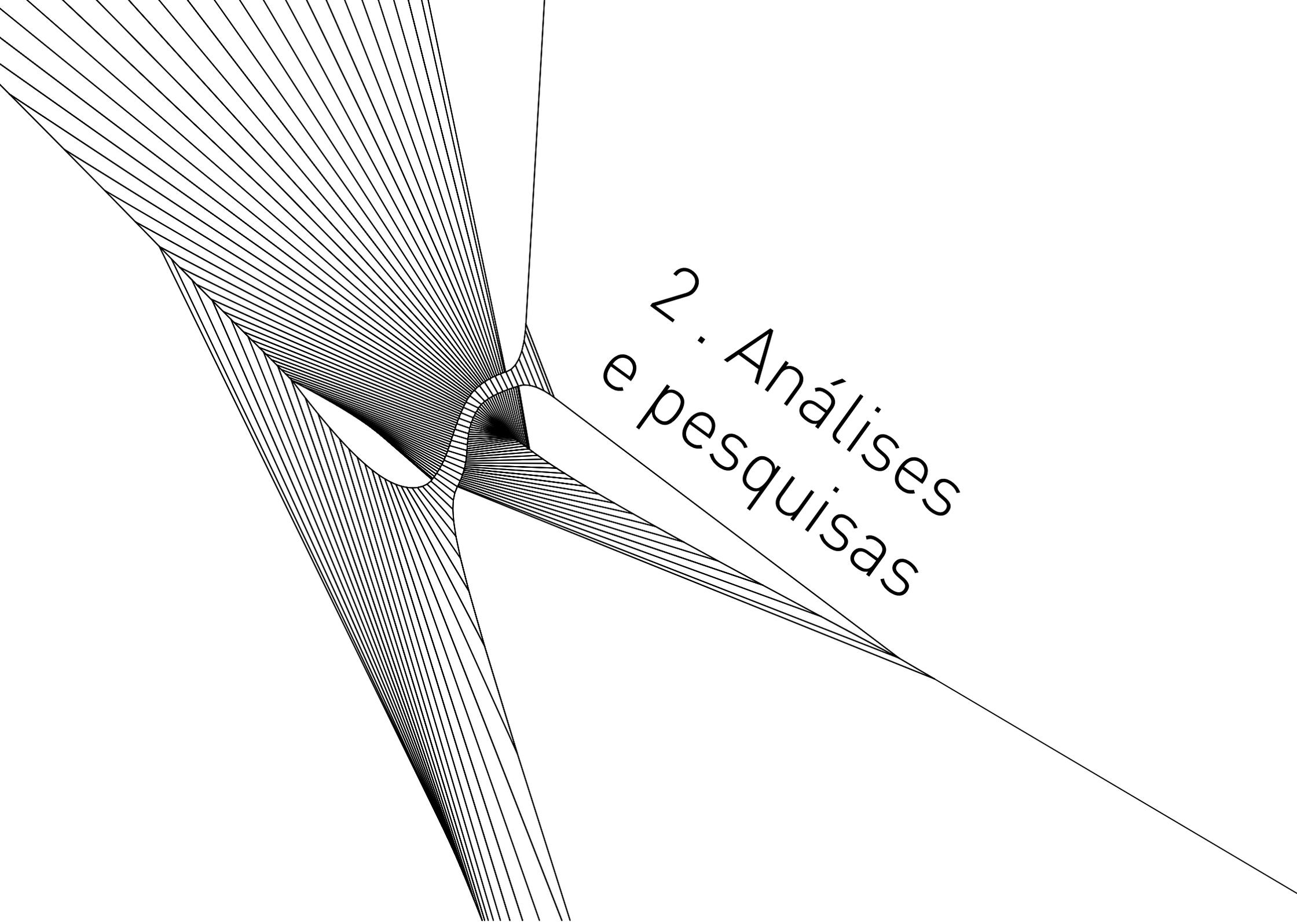
## 1.4 Metodologia

A metodologia deste projeto consistiu no desenvolvimento de uma série de experimentos, representados na imagem ao lado por círculos verdes tracejados. Eles são mostrados desta forma, pois não são fechados, recebendo a influencia de elementos externos como os testes com usuários. Cada experimento forma um conjunto de elementos, representados pelos círculos amarelos; Como exemplo os elementos do primeiro experimento foram processing, 3D, integração, arduino...

Depois que o experimento se encerra e chegamos a um resultado, os elementos mais pertinentes são passados para o próximo; Assim sucessivamente gerando um fio condutor que percorre todo o processo. Esse método pode ser executado infinitamente, mas como o projeto tem a demanda de produzir um resultado final o processo é interrompido, representado pela linha azul.

Assim ele passa a receber a influencia apenas dos testes com usuários e evolui com os elementos que herdou.



An abstract graphic composed of numerous thin, overlapping lines that create a sense of depth and movement. The lines are arranged in a way that suggests a three-dimensional structure, possibly a stylized letter or a complex geometric form. A central area where the lines converge is shaded dark, creating a focal point. The overall effect is that of a dynamic, layered composition.

## 2. Análises e pesquisas

## 2.1 levantamento de dados

Foram consultados livros e teses de Artes e design, para serem encontrados conteúdos e correlatos com documentação consistente a fim de embasar este projeto.

### Design de interação e computação pervasiva

A tese investiga os diferentes mecanismos tencionais envolvidos na utilização de sistemas de informação ambiente. Para tanto, conceitua e delimita a “computação pervasiva”( ... )Apresenta a evolução do design de interação. explicitando sua relação com o projeto de mídias interativas.

### Livro Art of the digital age

Esta publicação fala sobre o impacto das novas tecnologias nos meios de produção artística tradicionais como pintura e escultura. Reunindo mais de cem artistas do mundo todo, subdividindo seus trabalhos em novas classificações artísticas, como instalações, imaginação digital, animação por software e net art.

### Livro yourself sensing

Livro que reúne vários trabalhos artísticos e instalações que tem como tema a percepção de mundo e os sentidos humanos.

### DigitalbyDesign:CraftingTechnology for products and environments

É uma coletânea de trabalhos de visionários que trabalham com a relação entre a tecnologia e os produtos, por meio de abordagens de design e ambientes imersivos.

### livro talk to me: design and the communication between people and objects

Este livro aborda o lado subliminar dos produtos e como eles se comunicam conosco tanto direta quanto indiretamente. Estes objetos são mostrados em uma cronologia que vai de 1985 até os dias de hoje.

## 2.2 Pesquisa teórica

### Design de interação

“... O uso cada vez mais intenso dos computadores neste início de século tem provocado profundas transformações em diferentes esferas do cotidiano, estendendo-se da economia global às atividades particulares dos indivíduos. A dita revolução informacional, de fato, modificou sensivelmente a maneira como a sociedade se organiza, como as pessoas se relacionam e trabalham, alterando inclusive as noções de tempo, espaço, público, privado, coletivo, individual.

Os efeitos dessas transformações também podem ser percebidos no campo do Design. A partir da introdução da tecnologia computacional, além da modificação dos processos de produção, ocorreu também o surgimento de uma nova área de atuação para os designers. Se num primeiro momento os designers eram apenas mais uma categoria profissional que teve à sua disposição uma nova ferramenta de trabalho, ao longo dos anos passaram não só a utilizar computadores para realizar seus projetos, como também se envolveram com a

criação das próprias interfaces que fazem a mediação entre os sistemas computacionais e o homem. Essa área de atuação passou a ser conhecida como design de interação.”(PINHEIRO, 2011, p.76)

Este projeto está intimamente relacionado aos conceitos do design de interação, uma vez que buscamos não somente utilizar as ferramentas que nos são oferecidas, mas também explorar novas formas de comunicação, através da criação de nossas próprias ferramentas de trabalho. Adequando os instrumentos a atividade a ser executada e inovando a interação e a experiência do usuário.

Atualmente recursos de software, como a linguagem de programação processing, e a plataforma de prototipagem de hardware arduino, nos ajudam a desenvolver esse trabalho. Pois se tratam de ferramentas “open source” criadas para que usuários com pouco conhecimento de programação e eletrônica, possam desenvolver suas próprias ferramentas de interação.

Segundo o teórico, Gui Bonsiepe “O design é uma atividade intimamente relacionada com a criação de interfaces - entendidas aqui como elementos que permitem a interação entre um indivíduo e um objeto ou signo, visando à realização de uma ação efetiva. (BONSIEPE, 1997, p.12)”(PINHEIRO, 2011, p.77)

## Arte generativa

Segundo Philip Galanter “Arte Generativa se refere a qualquer prática artística onde o artista cria um processo, como um conjunto de regras de linguagem natural, um programa de computador, uma máquina, ou qualquer outro invento, que depois é colocado em movimento com um certo grau de autonomia contribuindo ou resultando em uma obra de arte completa.”(GALANTER, p.4)

Como exemplo deste tipo de artista temos Casey Reas, um dos criadores da linguagem e ambiente de desenvolvimento Processing. Seu trabalho é baseado no conceito de draw with code, onde seus desenhos são criados a partir de um script, que trabalha de forma autónoma.

Arte generativa pode ser empregada neste projeto para ampliar a interação do usuário com sua modelagem. Este tipo de arte é normalmente feito com a entrada de um parâmetro e a partida é dada no programa, encerrando a interação nesse momento. Se o usuário pudesse intervir a todo o momento com um determinado controle sobre as transformações generativas, seria um avanço com relação às interfaces existentes.



1. Obra de abertura do vídeo How To Draw With Code, 2012



## Biomimética

A impermanência é um dos conceitos que norteiam a Biomimética. Todos os animais são seres impermanentes, pois as condições externas do ambiente em que vivem são suscetíveis a mudanças e elas podem influenciar no seu crescimento ou no desenvolvimento de certas habilidades, que em outro ambiente permaneceriam latentes.

Já os computadores e outros equipamentos digitais são imutáveis, permanentes, porque eles não absorvem inputs externos e não reagem a eles. Essa propriedade característica dos computadores tem seus prós e contras, uma vez que a falta de autonomia torna as reações do computador previsíveis, eles não podem surpreender aqueles que conhecem bem seu sistema. Por outro lado, essa previsibilidade os faz indefesos no que tange a sua parte física e frustrantes no sentido de saída de dados, tendo em vista sua impossibilidade de responder a estímulos externos.

Uma possível inovação que resolveria o problema de sua previsibilidade seria a Web 3.0, onde uma busca na internet não seria tratada por palavra separadamente, mais o buscador procuraria entender o sentido da sua frase e buscar o conceito em sites.

Existe um ramo da Biomimética polêmico que estuda a possibilidade de uma união entre seres vivos e partes robóticas e computacionais, visando atingir um meio termo entre a previsibilidade dos segundos com a maleabilidade e a capacidade de resposta a estímulos que os primeiros possuem.

Dentro deste universo o projeto será focado nos conceitos e aprendizados que podemos extrair dos organismos vivos para tornar as ferramentas de modelagem tridimensionais mais inteligentes; por um lado fazendo com que elas compreendam melhor a nossa entrada de dados, gerando modelos mais fieis ao que desejamos, e por outro, pode gerar formar imprevisíveis, que até poderiam ser consideradas errôneas ou defeituosas que serviriam de inspiração para o trabalho, mas de forma controlada onde o usuário escolhe quando e com que intensidade ativar isso.

## High-tech Low-tech

High-Low Tech, é um grupo de pesquisa do MIT Media Lab, que inaugurou o conceito de high-tech low-tech. Seu objetivo é democratizar a tecnologia, tornando-a simples e barata, com o uso de elementos presentes em nosso cotidiano. Para eles, no futuro, não seremos mais tão dependentes da indústria, pois criaremos nossos próprios dispositivos. Chegando ao ponto da intersecção entre tecnologia e artesanato.

Um indivíduo, mesmo analfabeto, é capaz de criar ou concertar ferramentas que o ajudam no dia a dia, como é o caso de muitos trabalhadores rurais. Eles conseguem fazer isso, pois tem acesso às peças que compõem sua ferramenta, sejam comprando ou reaproveitando de outro objeto, e compreendem, mesmo que de maneira básica, a composição e montagem de tal equipamento.

Para o high-tech low-tech uma vez que compreendemos alguns princípios básicos de como os equipamentos eletrônicos funcionam, podemos usar seus componentes somados a materiais de fácil aquisição como tecidos, papel laminado, madeira... E criar aparelhos que atendam a necessidades específicas de cada um.

Esse movimento é reforçado pelo desenvolvimento de ferramentas que facilitam a utilização de componentes eletrônicos, como é o caso das plataformas Arduino e Microsoft Gadgeteer e pelo barateamento da computação, como é o caso do computador raspberry pi que custa apenas 25 dólares.

## Haptic

Haptic, é o nome dado a toda tecnologia que possibilita interação e feedback físico, sensorial entre homem e máquina. Esse tipo de equipamento é largamente utilizado na indústria dos games, para ampliar a imersão do jogador. Um exemplo simples é o equipamento “Rumble Pak”, para o console Nintendo 64, ele trabalhava vibrando o controle toda vez que um estímulo é necessário. Um mais complexo é o controle Novint falcon que permite ao usuário sentir a textura dos modelos e o coice de armas.

Não nos damos conta, mas lidamos a todo o momento com este tipo de interface. Por exemplo, quando tocamos em uma tela de celular e ela responde a este estímulo modificando o que é mostrado e, em alguns casos, vibrando para ampliar a sensação de toque.

## 2.3 Pesquisa de similares

### Sketch Furniture

Este trabalho busca tornar esboços de mobílias em peças reais, por meio da captura de movimento e da prototipagem rápida. Com o uso de um led na ponta de uma caneta o usuário consegue fazer o desenho do objeto no ar, com as dimensões desejadas. Seus movimentos são captados por quatro câmeras, que geram o desenho de uma linha tridimensional contínua. Posteriormente este traçado é convertido em uma malha, ganhando assim volume e possibilitando sua prototipagem.

A saída do arquivo é feita por uma impressora 3D a laser. Que consiste em aplicar os raios sobre camadas de um gel especial, que enrijece pela passagem da luz. O resultado final é muito fiel ao que é visto no computador, com formas lisas e sinuosas muito características desse tipo de máquina.



2 . Art basel Miami, With Barry Friedman Gallery LTD, 2005



3 . Objetos prototipados com uso de laser

## Artisan electronique

Neste projeto de pesquisa, realizado em parceria com Tim Knapen e RepRap community, os designers combinaram técnicas de cerâmica tradicionais com os novos meios digitais. Nele o usuário interage com um torno virtual para gerar formas orgânicas.

A instalação composta por uma tela que mostra um cilindro em malha do tipo mesh, este gira em sentido horário continuamente. Logo a sua frente encontra-se um sensor de distancia a laser que percebe a posição da mão. Assim como em um trono para argila, quando a mão do modelador se aproxima das laterais do objeto, o volume de “massa” começa a ser subtraído neste local. O usuário pode recomeçar do zero quantas vezes quiser, até decidir que o trabalho está pronto. Posteriormente os modelos são prototipados em uma impressora 3D de deposição, modelo RepRap modificado. Originalmente esta máquina trabalhava estrudando plástico derretido como ABS, mas nesse caso ela foi alterada para trabalhar com argila.

Os modelos impressos são facetados devido a sua herança da malha mesh, o que é um claro exemplo de sua transposição do meio digital para o analógico.



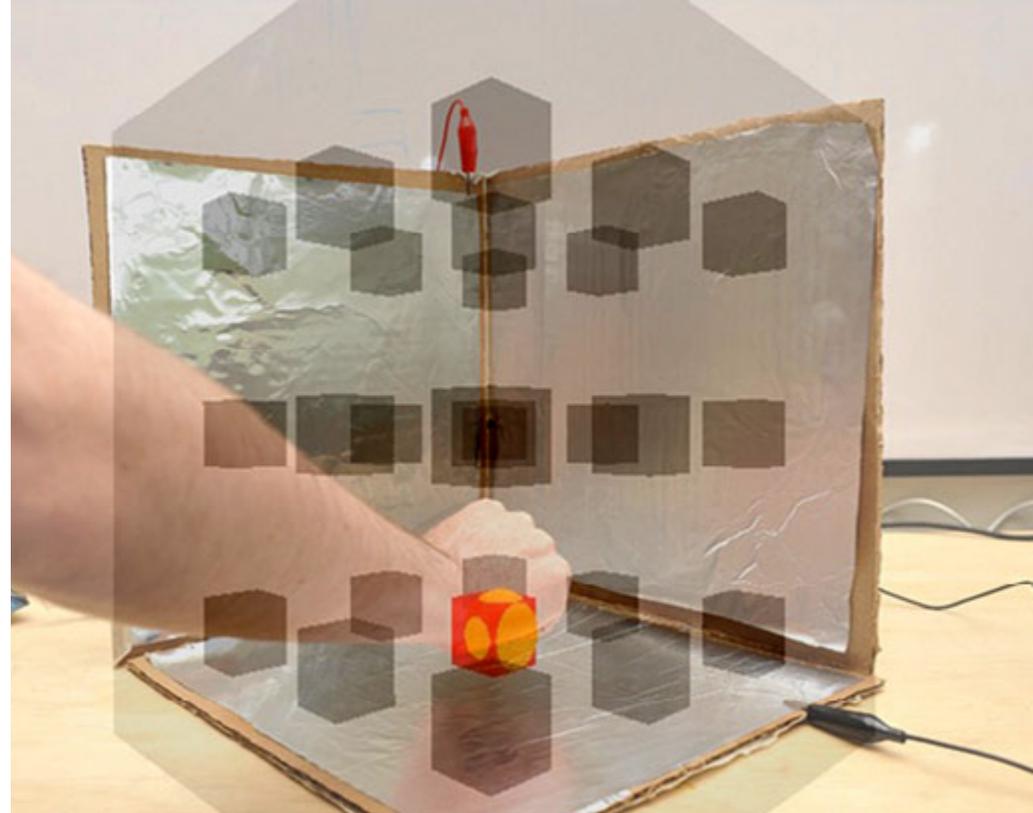
4 . Artisan electronique, instalação realizado em parceria com Tim Knapen e RepRap community, 2010

## 3D Tracking Interface

Projeto que segue os preceitos do “high-tech low-tech”. Nele o pesquisador Nick Normal criou um sistema que captura a posição da mão do usuário dentro de um espaço limitado. Para isso, ele usou materiais convencionais como papelão e papel laminado, construindo de maneira artesanal uma interface física que se comunica com o computador através de uma Arduino.

A lógica deste sensor encontra-se na formação de três campos eletromagnéticos, representando os eixos X, Y e Z, por meio das chapas de papel laminado. Tais campos sofrem alteração pela presença da mão do usuário, por meio da leitura dos valores dos campos é possível descobrir a posição da mão.

Os resultados obtidos não são muito precisos, pois o aplicativo só reconhece 27 cubos no espaço, quando a mão se movimenta de uma localidade para outra a ação é mostrado de forma interpolada como uma animação. Mas continua sendo uma ferramenta válida para criar dispositivos de entrada, uma vez que seu custo é baixo e gera imersão.



5. 3D Tracking Interface,  
Nick Normal, 2010

## Novint falcon 3D/ CRE8

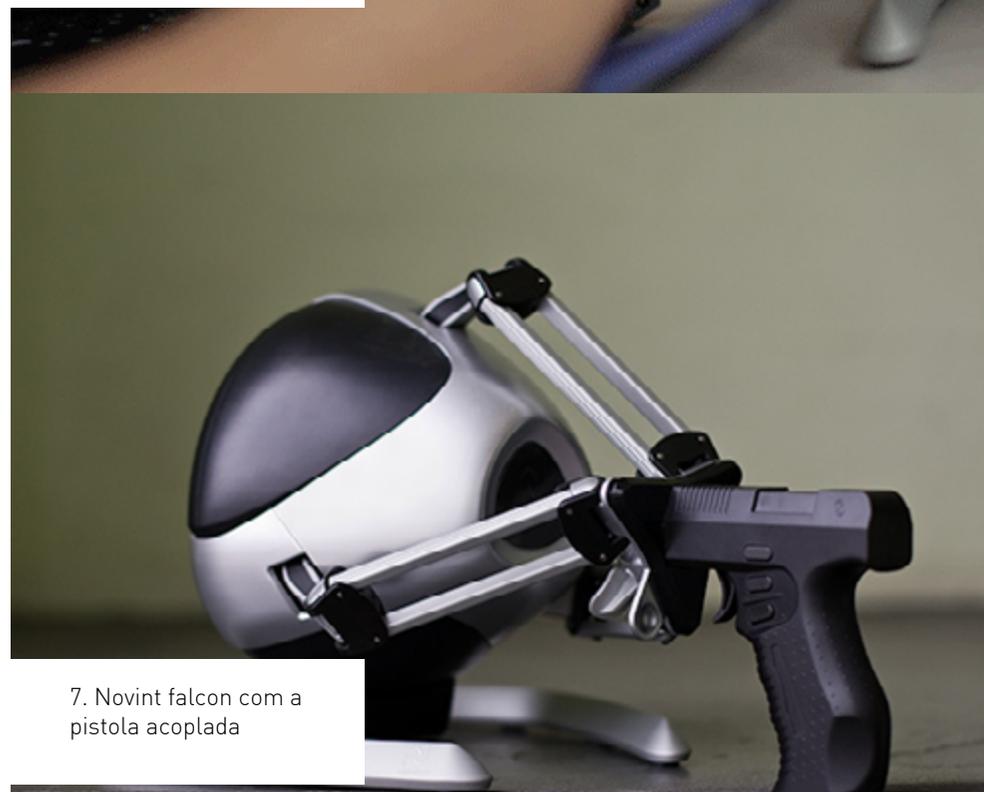
É um mouse que trabalha em três eixos, que por meio da aplicação de forças gera a sensação de toque em modelos virtuais. Seus motores executam movimentos tão precisos que é possível perceber, com nitidez, a textura de tais objetos. Originalmente foi desenvolvido como controle de “feedback” físico para jogos, mas acabou encontrando espaço em outra áreas, inclusive sendo usado para treinar cirurgiões.

CRE8 é uma aplicação de modelagem gratuita, para esta interface haptic. Sua curva de aprendizado é muito rápida, em poucos minutos uma pessoa que nunca teve contato com modelagem pode alcançar resultados satisfatórios. Basta que o usuário crie uma forma primitiva no ambiente de trabalho e pegue uma ferramenta de edição. A modelagem é feita esticando e comprimindo partes do objeto criando deformações, a sensação assemelha-se com a manipulação de uma massa plástica.

O programa trabalha com malha do tipo mesh, tornando se uma boa alternativa para a modelagem de personagens, mas infelizmente ele não conta com muitas opções de formatos para exportação.



6. software CRE8 em uso.

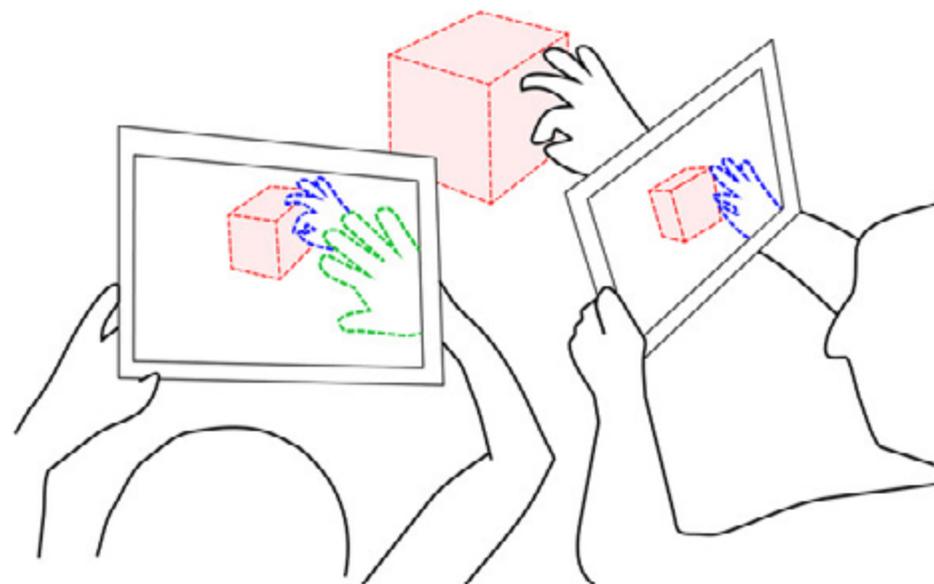


7. Novint falcon com a pistola acoplada

## T(ether)

Este é um projeto do MIT Media Lab. Os pesquisadores desenvolveram um aplicativo para Ipad, que torna possível a interação com modelos virtuais por meio da realidade aumentada. O aparelho serve como uma janela para o mundo virtual, onde usuário deve segurá-lo com uma das mãos e com a outra, por traz da tablet e vestindo uma luva especial, passa os comandos que deseja, neste momento a mão do operador aparece na tela como um modelo virtual. A luva mencionada é composta por sensores costurados nas pontas dos dedos e o clique é feito com o polegar opositor. Cada dedo acionado executa uma ação, exceto o mínimo. Com o médio são criados o objetos, o usuário pode escolher qualquer forma primitiva, com o indicador eles são movidos e rotacionados, pelo anelar o objeto alvo da ação é deletado.

A sensação de tridimensionalidade é gerada pela sincronização da imagem com a posição da cabeça do manipulador, ou seja este pode caminhar ao redor e entre os objetos criados que eles permanecerão no mesmo lugar do espaço. Outro ponto forte do programa é o módulo de modelagem colaborativa. Nele, outro usuário com os mesmos equipamentos pode interferir na modelagem de terceiros em tempo real.

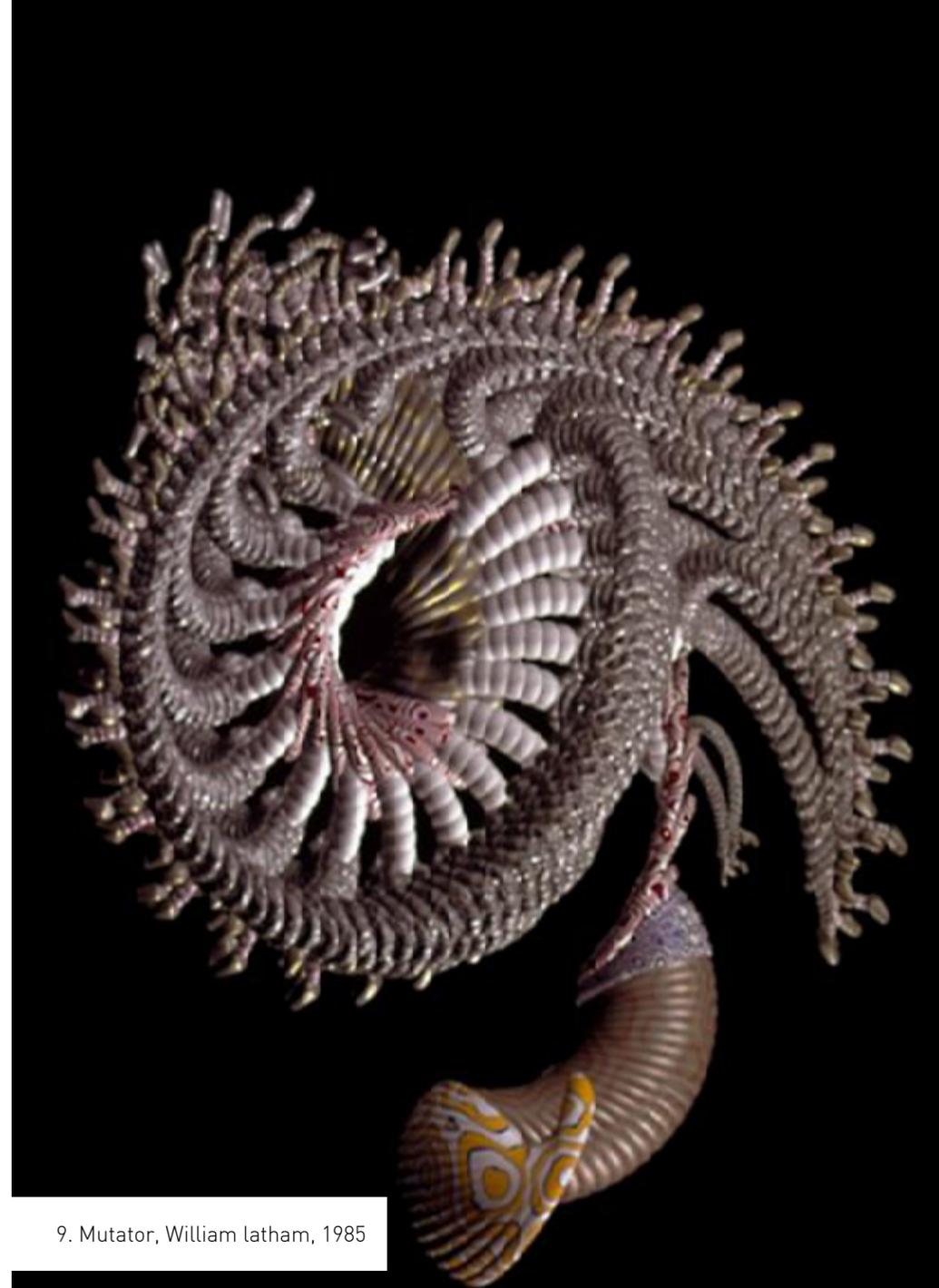


## Mutator

William Latham foi um dos pioneiros no trabalho com modelagem tridimensional e animação em 1985, tendo participado da equipe do centro de pesquisas Xerox park.

Neste trabalho, Latham, baseando-se no processo de seleção natural, cria uma animação onde nove objetos tridimensionais idênticos surgem, entrando em um processo de mutação. Passado algum tempo cada um adquire características próprias até que um é escolhido pelo usuário para permanecer na tela enquanto os demais desaparecem. Todo o processo recomeça, então, com o objeto escolhido se multiplicando por nove e sofrendo mutações diferentes. Isso se repete até que o usuário se dê por satisfeito.

Através deste estudo William leva para o ambiente da modelagem virtual as questões levantadas por Darwin, onde o mais forte sobrevive. No caso o mais forte é o que mais se assemelha a intenção do artista.



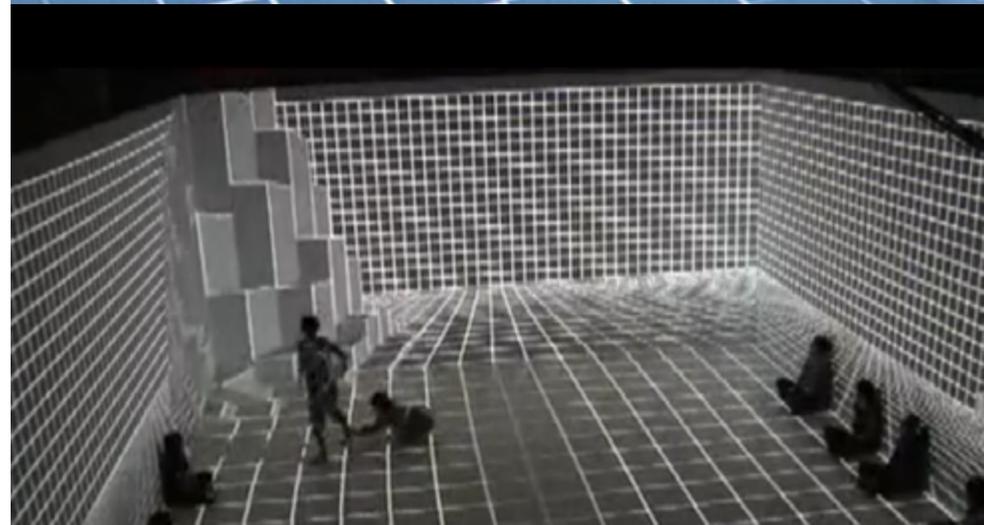
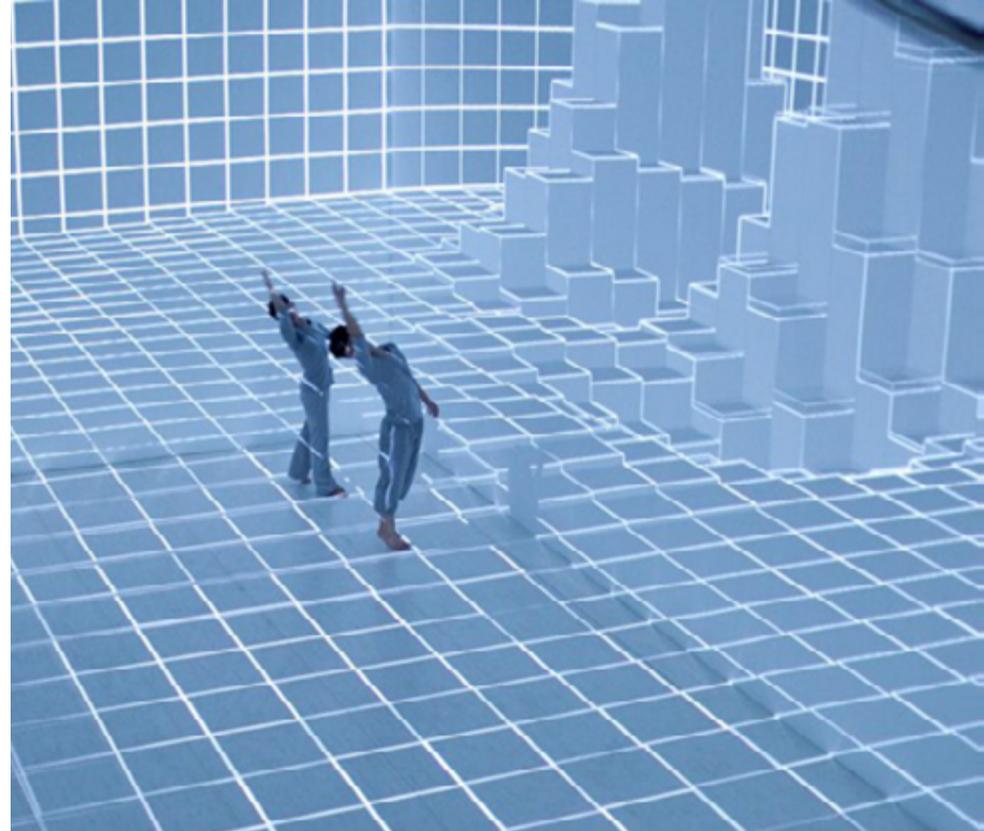
9. Mutator, William latham, 1985

## Seventh Sense (Excerpt)

É um espetáculo de dança contemporânea, realizado pelo grupo Anarchy dance theatre em parceria com UltraCombos, composto por um cenário que reage em tempo real aos movimentos dos dançarinos. Em cada um dos atos os feitos visuais variam, indo desde geradores de partículas á manipulação de um modelo 3D que extrapola os limites físicos do palco.

O projeto foi desenvolvido em “openFrameworks”, uma biblioteca de código aberto baseada em C++. Por meio dela, quatro câmeras foram usadas para capturar a silhueta e a posição dos dançarinos. Posteriormente esses dados geram as animações que por meio de quatro projetores são vistas nas paredes do palco. O som também é gerado em tempo real usando o software MAX/MSP.

A importância desse correlato está na percepção do corpo como um todo, transmitindo dados para o computador, informações estas que transformam uma estrutura tridimensional não com o intuito de gerar um objeto, mas que futuramente podem ser aplicados para esta função, outro aspecto está na completa imersão tanto do público quanto do usuário, como podemos ver nos vídeos e imagens parecem ter sido transportados para outra dimensão.



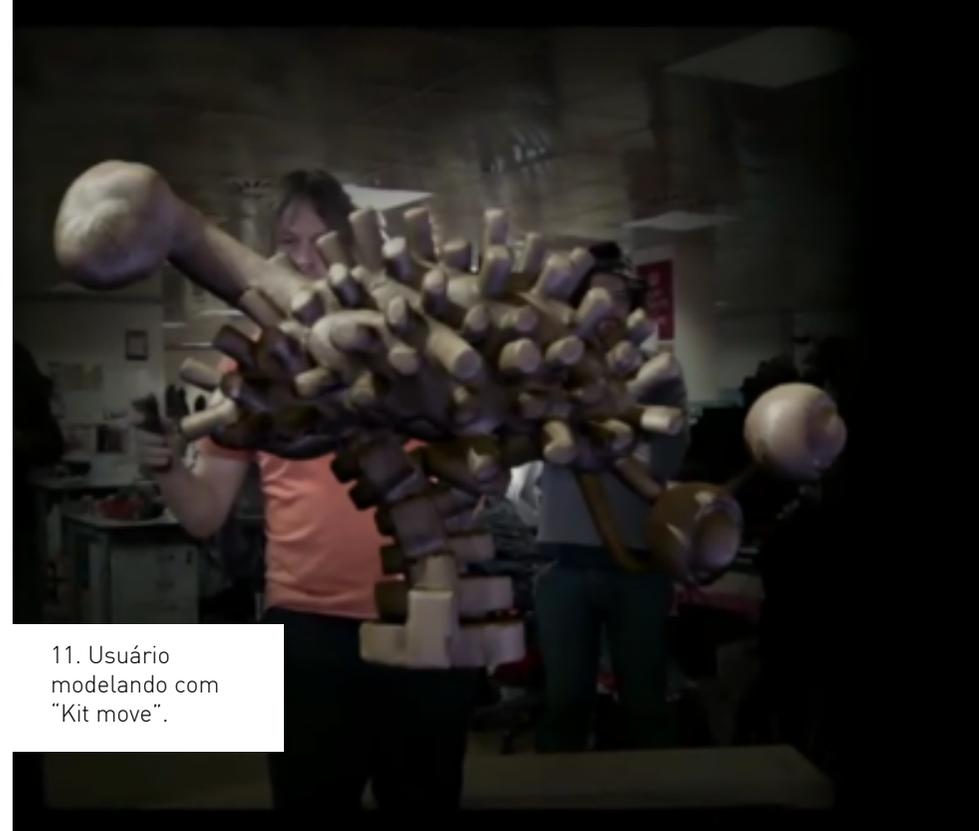
10. Espetáculo de dança Seventh Sense (Excerpt)

## Media Molecule's PS4 Tech Demo

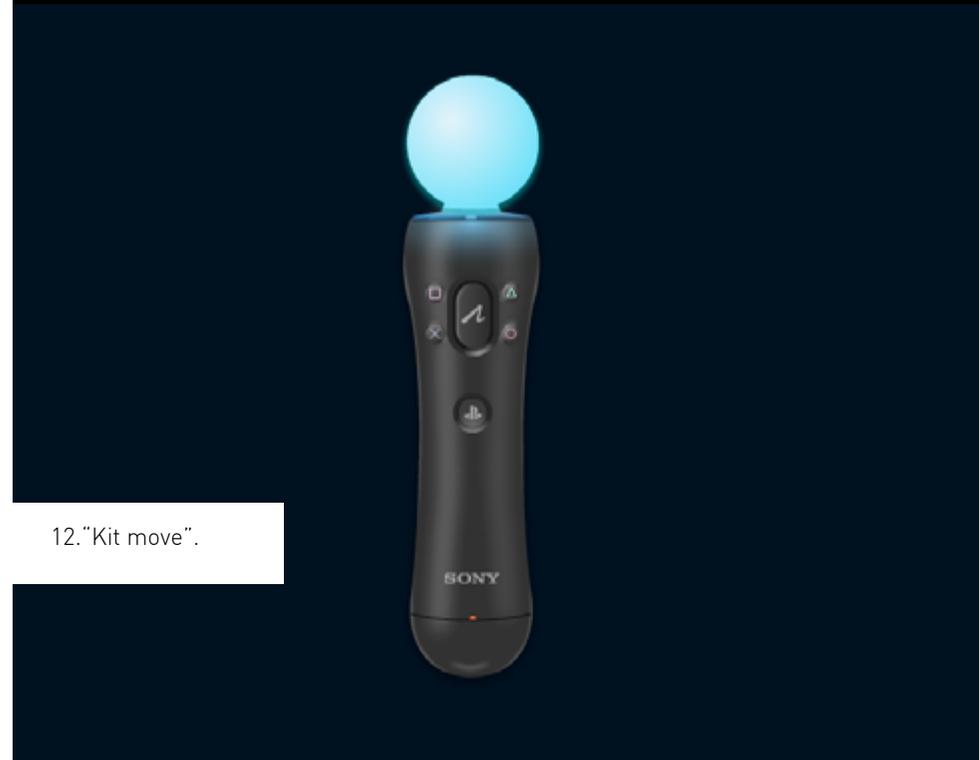
Tech Demo é uma demonstração de uma futura tecnologia que ainda está em fase de desenvolvimento. Muitas vezes o protótipo ainda não está concluído para ser exibido então são feitas simulações com o intuito de apresentar a aplicação para investidores ou como "teaser" para futuros consumidores.

Este Tech Demo foi desenvolvido pela Media Molecule para a Sony com o intuito de apresentar uma futura implementação para o controle PS4, com o uso do controle 3D "kit move". Nesta aplicação o jogador poderá criar personagens, objetos e senários para seus jogos, como se estivesse fazendo uma modelagem em Clay. Posteriormente esses modelos poderão ser compartilhados entre outros jogadores.

Este exemplo torna o meu projeto ainda mais pertinente, uma vez que, mostra uma tendência da indústria em abordar a interação do usuário com modelos 3D. Neste caso o foco é baseado no universo lúdico dos games, onde buscam gerar uma forma simples para um jogador gerar modelos tridimensionais.



11. Usuário modelando com "Kit move".



12. "Kit move".

## 2.4 Pesquisa estética

A estética do projeto está ligada com os atuais movimentos de arte eletrônica e interativa, onde os aspectos visuais estão atrelados à materialidade e não existem grandes preocupações em mascarar a matéria prima, tanto a virtual quanto a real.

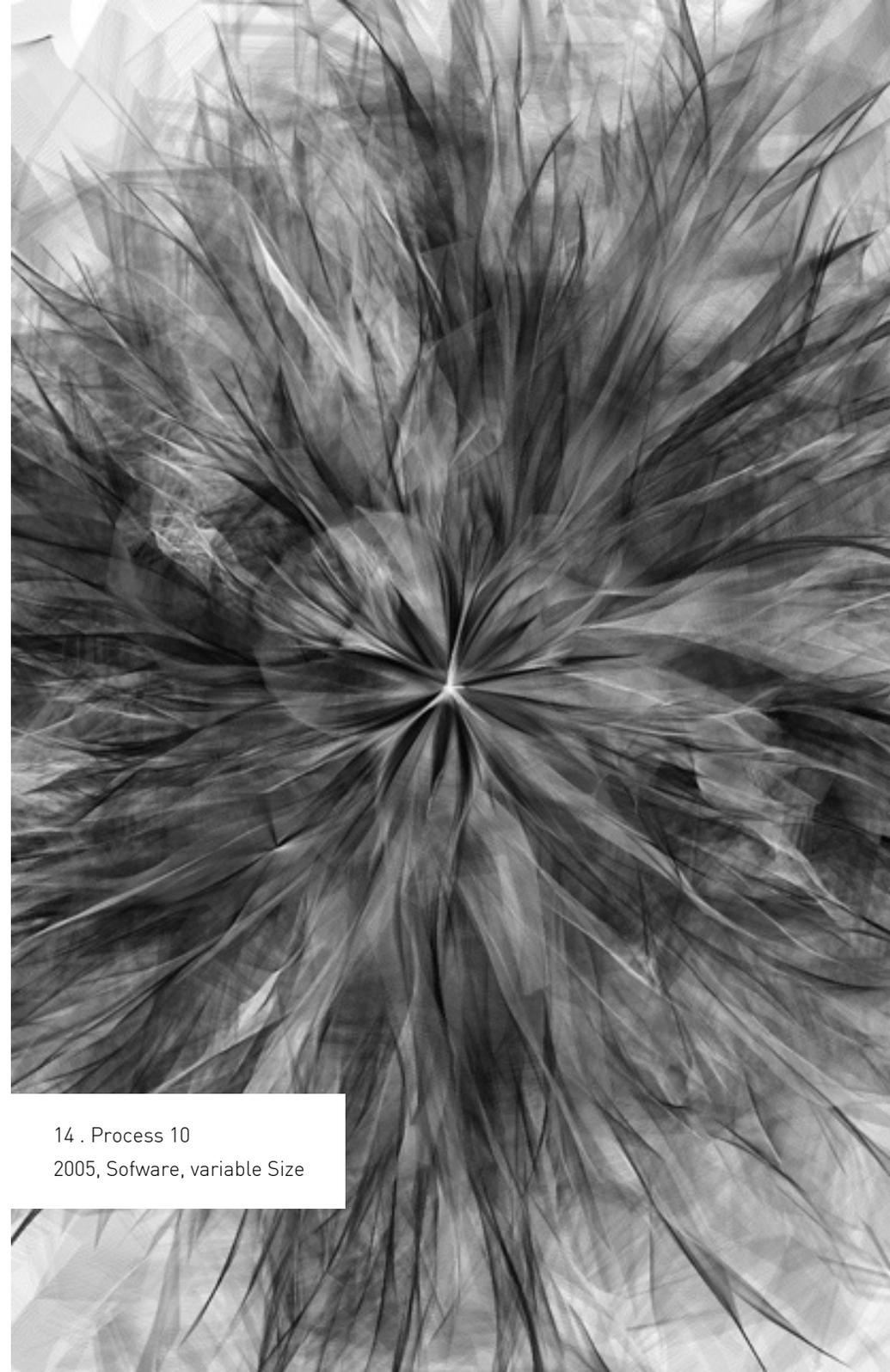
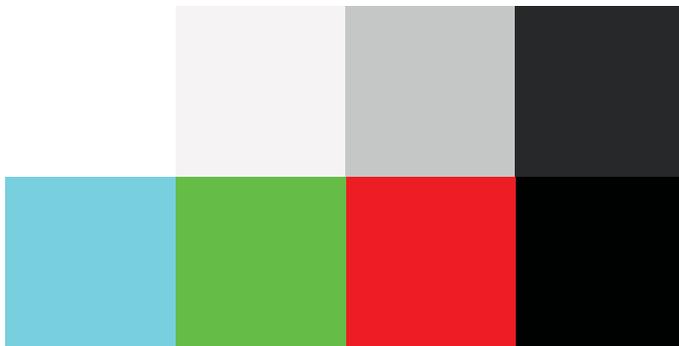
Os diagramas em tela são simples. Usa-se uma grande quantidade de linhas, formas geométricas básicas e wireframes à mostra, sem a preocupação de utilizar renderes realistas. Isso é feito para concentrar todo o processamento da máquina na atividade programada e também para abordar apenas o conceito. Com relação a paleta de cores, percebemos o uso de tons pastéis e tonalidades de cinza quando se tratam de objetos, já em instalações interativas, com representação em tela encontramos cores vivas e saturadas.

A aparência física dos objetos é normalmente livre de tinta ou massa para lixar, deixando clara a forma de fabricação. As técnicas de produção mais usadas são as de prototipagem rápida como cortes a laser, impressoras por deposição e fresadoras.



13. Artisan electronique, instalação realizado em parceria com Tim Knapen e RepRap community, 2010

Analisando o trabalho de Casey Reas encontrado ao lado, podemos observar o uso de inúmeras linhas orgânicas pretas que são dispostas em áreas de saturação de elementos, contrapondo-se com regiões de menor concentração gerando diferentes tons de cinza. Partindo desta obra, retirei parte de minha paleta de cores. Esta é composta por preto, branco, gradações de cinza. As demais cores possuem ligação com a estética flat de design que usa cores saturadas como uma de suas linguagens. A questão das linhas e “grids” foi incorporada no projeto por meio do diagrama de voronoi, com ele posso gerar elementos gráficos bi ou tridimensionais compostos apenas por linhas, fazendo uma menção a estética computacional. Ao mesmo tempo, cria um contraste por se tratar de uma estrutura presente na natureza.

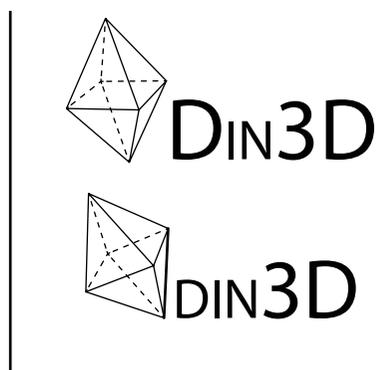


14 . Process 10  
2005, Software, variable Size

# Estudos de logos

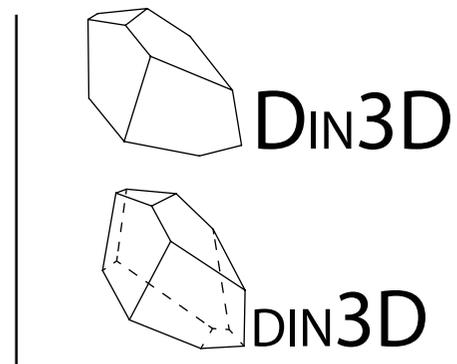
Este projeto chama-se Din3D, que significa design de interação tridimensional. No momento do desenvolvimento das propostas de logo, tínhamos em mente que a marca deveria ter algum elemento 3D. Surgindo os primeiros estudos onde colocamos um cubo distorcido junco ao nome do projeto.

Din3D - Font Myriad Pro, cubo



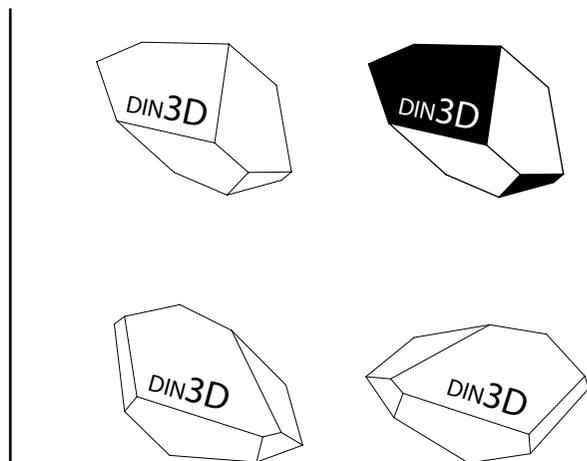
Todos os grafismos do projeto foram feitos por meio de estruturas de voronoi bidimensionais, com o intuito de estabelecer uma linguagem visual única, substituímos o cubo por uma célula voronoiica.

Din3D - Font Myriad Pro, ao lado de uma célula voronoiica



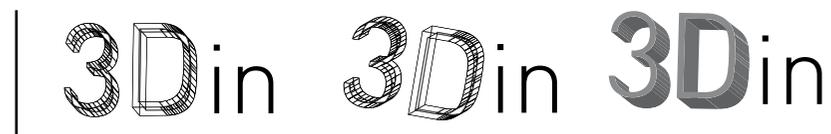
Neste momento buscamos uma maior integração do nome com o desenho, para isso geramos possibilidades de encaixe do nome dentro de células voronoicas.

Din3D - Font Myriad Pro, dentro de uma forma voronoica

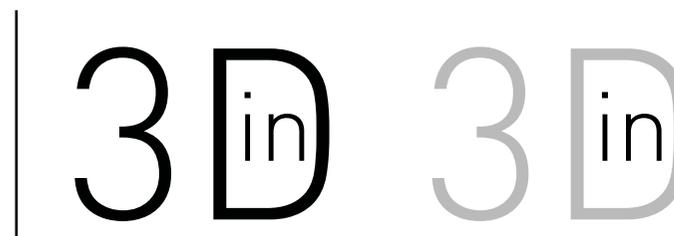


Ouve um momento que cogitamos chamar o projeto por 3Din ou 3D in, trazendo uma ideia de inclusão, mas esta possibilidade foi descartada, uma vez que este trabalho busca o movimento de saída do mudo virtual e não o aprisionamento neste.

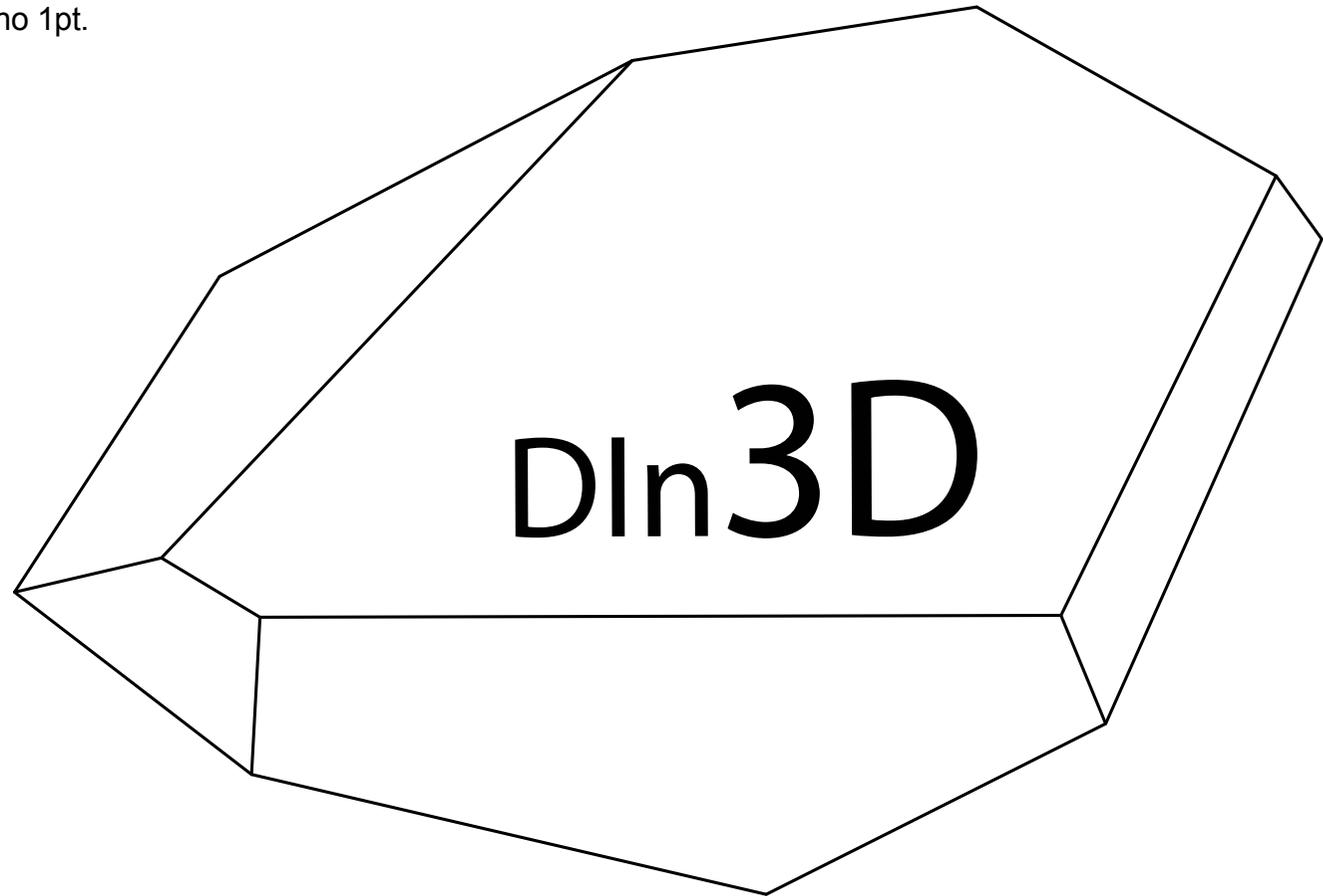
3Din - Font DIN light, 3D tridimensional



3Din - Font DIN light



Esta foi a logo escolhida para representar o projeto por guardar uma boa proporção entre fonte e desenho. O voronoi possui como característica a leveza visual, pois é matematicamente harmonioso. Tipografia usado Myriad Pro, stroke do desenho 1pt.



## 2.5 Pesquisa técnica e tecnológica

### Processing

“Processing é uma linguagem e ambiente de programação, “open source”, para pessoas que desejam criar imagens, animações e interações. Inicialmente desenvolvido para servir como um caderno de esboços de software e para ensinar fundamentos de programação dentro de um contexto visual. Devido a seu sucesso, processing acabou tornando-se uma ferramenta de uso profissional completa. Hoje, há dezenas de milhares de estudantes, artistas, designers, pesquisadores e entusiastas que utilizam processing para a aprendizagem, prototipagem e produção.”(tradução livre do inglês)

O projeto teve início em 2001 por Casey Reas e Ben Fry, ambos foram pesquisadores do MIT Media Lab no grupo de computação. Processing é uma API de java, ou seja quando um programa é escrito em processing na verdade o arquivo executado é um java. Isso ocorre pois a linguagem de base tem muito poder para gráficos e bom desempenho, mas não é muito intuitiva para usuários com pouco conhecimento de programação.

### Arduino

“Arduino é uma plataforma “open-source” de prototipagem eletrônica baseada em um hardware flexível atrelado a um software fácil de usar. É destinado a artistas, designers, amadores e qualquer pessoa interessada em criar objetos ou ambientes interativos.”(Tradução livre do inglês)

O funcionamento da placa é simples. Nela podemos conectar sensores digitais ou analógicos, servindo como entradas de dados, assim como motores, LEDs e outros componentes, como saída. Tudo isso é coordenado por um ambiente de programação, onde o usuário pode tratar os dados recebidos e as respostas da placa. A plataforma de programação dá suporte a linguagens como: C, C++, java e processing. Sendo a última sua linguagem principal.

Projeto iniciado em 2005 na cidade de Ivrea, Itália, com o intuito de ser uma placa de prototipagem de circuitos mais barata que os modelos existentes o momento. Para tornar isso possível, arduino foi baseada em um chip Atmel AVR de 8bits, que é bastante poderoso por conter uma memória Flash interna, o que retira a necessidade de um sistema externo de armazenamento para os comandos. Devido a isso podemos dizer que a

arduino é um sistema de computações embarcada.

O sistema pode funcionar de duas formas “standalone” (autossuficiente) , onde a placa opera sem o auxílio de um computador, podendo tornar-se ainda mais independente se usar uma bateria. Ou trabalhar em conjunto com uma aplicação dentro de um computador, servindo como ponte entre o mundo virtual e o físico.

Por se tratar de um projeto “open-source” , Qualquer pessoa pode desenvolver sua própria versão do circuito, o que gerou o surgimento de inúmeras variações cada uma focada em uma aplicação específica. Como exemplo temos:

**Arduino Severino:** placa de baixo custo de produção, desenvolvida por um brasileiro. Usa uma porta serial no lugar de um USB.

**Arduino Leonardo:** Usa um chip ATmega32u4, possibilitando que a placa seja reconhecida pelo computador como um hardware estemo comum, assim como um mouse ou teclado.

**LilyPad Arduino:** Menor arduino existente, ela mede aproximadamente 50 mm de diâmetro e tem 3 mm de altura. Foi feita para dar suporte a projetos de tecnologia vestível, podendo ser costurada diretamente na roupa e ser lavada.

As placas mencionadas anteriormente não foram desenvolvidas pelo arduinoteam( grupo que criou o sistema), mas são consideradas oficiais por eles. Destes temos a arduino duemilnove e a UNO, Ambas são quase idênticas variando apenas o chip em que foram baseadas.

## Microsoft Gadgeteer

É a plataforma de prototipagem de hardware da Microsoft. Na placa arduino o usuário precisa ter mesmo, que muito superficialmente, algum conhecimento de eletrônica para poder conectar corretamente os componentes. No caso do Gadgeteer todo o componente vem em forma de pequenas placas que são encaixadas com pinos apropriados, isso dispensa o uso de solda e torna o trabalho mais intuitivo. Um de seus pontos fortes está na facilidade para usar telas sensíveis ao toque, bastando apenas acoplar o módulo a placa. Por outro lado esse sistema é menos flexível deixando o usuário preso as peças desenvolvidas pela Microsoft.

## Kinect / openNI

Kinect é um controle desenvolvido para o console Xbox 360. Basicamente é uma câmera tridimensional que capta a posição e os movimentos dos jogadores. Ele foi criado para concorrer no ramo de jogos onde a interação é feita com o corpo. Essa demanda de mercado foi criada pelo console Wii da Nintendo, que com seu controle “wii mote” o usuário interagia com o jogo por meio de movimentos com a mão sacudindo o controle, já o kinect, busca ampliar este conceito transformando o corpo todo em um controle.

O aparelho funciona emitindo um laser ultravioleta que é disperso por um aparato ótico, todos os objetos atingidos são captados por uma câmera VGA específica para esse tipo de raio. Esta serve como um sensor 3D pegando a posição X,Y e Z de cada pixel, sua resolução é de 320x240 tendo de 1,2 m até 3,5 m de profundidade de campo. Ao lado deste sensor existe uma câmera RGB convencional com 640x480 de resolução, ela serve para captar texturas que podem ser usadas em programas de reconhecimento facial. O equipamento ainda possui quatro microfones que eliminam ruídos e

o eco, para um perfeito reconhecimento de voz( até o momento desta publicação o recurso só está disponível em inglês).

O kinect possui processador e software próprio, computando os dados dos sensores de forma independente do console ou computador que está ligado. Seu firmware é bastante poderoso tendo a capacidade de reconhecer a presença de 6 jogadores e o esqueleto de 2, ao todo são 48 articulações. Outro aspecto importante é que mesmo que a câmera não pegue todo o corpo ele consegue completar o esqueleto automaticamente. Isso é possível graças a um sistema que tem aproximadamente 200 poses mais comuns.

OpenNI , em uma tradução livre significa interfaces naturais livres. É uma biblioteca gratuita que permite a comunicação com o aparelho através de diversas linguagens de programação.

## Modelagem virtual

O conceito de modelagem virtual não está ligado apenas a um objeto dentro do computador, como podemos pensar, mas sim a objetos que não estão presentes no mundo real, ampliando esse sentido. Por exemplo, um desenho feito com geometria descritiva em uma folha de papel é considerado virtual.

O conceito da criação de um objeto virtual no computador, como conhecemos, é fácil e poderia ser feita escrevendo as coordenadas de seus vértices em um bloco de notas e usando um software que as compreenda da forma que foram escritas, gerando a imagem do mesmo. Todavia, se o objeto a ser criado for muito complexo a tarefa torna-se exaustiva. Para contornar esse problema os programas atuais e suas ferramentas foram criados para tornar essa tarefa mais ergonômica e simples.

Trabalharemos dois tipos de malha mais comumente usados: malha nurb e a malha mesh.

## NURBS

NURBS é uma sigla em inglês “Non Uniform Rational Basis Spline” ( splines base não uniformes racionais), ou seja são superfícies representadas por “splines”, que por sua vez são curvas de Bézier. Este tipo de curva representa as formas de maneira matematicamente perfeita sendo formada pela interpolação entre dois ou mais pontos. Esse tipo de malha tem sido usado no design de produto, por ser facilmente parametrizável o que acelera o processo de criação, e em processor de prototipagem pela sua fidelidade nas formas geradas.

## MESH

A malha mesh representa as superfícies por meio de pontos (vértices) que são ligados por retas ( edges) formando faces poligonais. Devido a isso ela não representa as curvas de maneira real, em modelos de formas orgânicas as partes arredondadas são representadas aumentando-se o número de iterações ( polígonos) causando a sensação de curvatura. Esse tipo de malha é largamente usado em aplicações gráficas como efeitos para filmes e modelos para jogos, devido a sua facilidade de renderização e por comportar bem animações.

## Rhinoceros

É um programa de modelagem tridimensional que trabalha com malhas do tipo NURBS. Originalmente ele foi criado como um plug-in de visualização para o AutoCAD da Autodesk, mas acabou evoluindo para um software independente levando características de seu programa base como o prompt de comando na parte superior e muitos atalhos de teclado.

Este é um dos programas de CAD mais utilizados no mundo devido ao seu baixo custo se comparado com a concorrência, suporte a vários tipos de arquivos vindo de outros programas servindo em muitos casos como conversor de formatos, interface intuitiva e de fácil aprendizado e ampla gama de plug-ins.

### “Biblioteca”

Quando se trata de programação, uma “biblioteca” é um conjunto de funções pré-programadas que ajudam o programador a desenvolver um código mais rapidamente, ou quando o código exige conhecimentos que o profissional não possui.

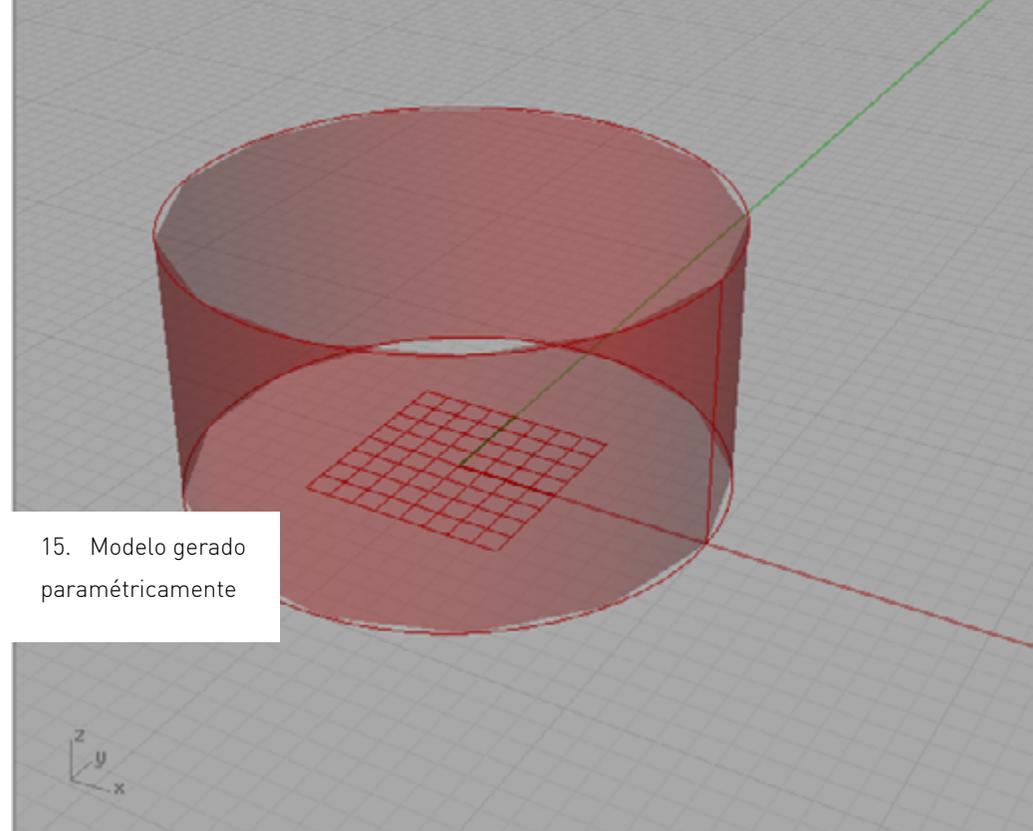
## Grasshopper

“Grasshopper é um editor de algoritmo gráfico integrado com ferramentas de modelagem do programa Rhinoceros 3D. Ao contrário RhinoScript ( linguagem de programação própria do software), Grasshopper não requer nenhum conhecimento de programação ou scriptagem, mas ainda permite que os designers construam formas gerativas de maneira simples e inspiradora.”( tradução livre do inglês).

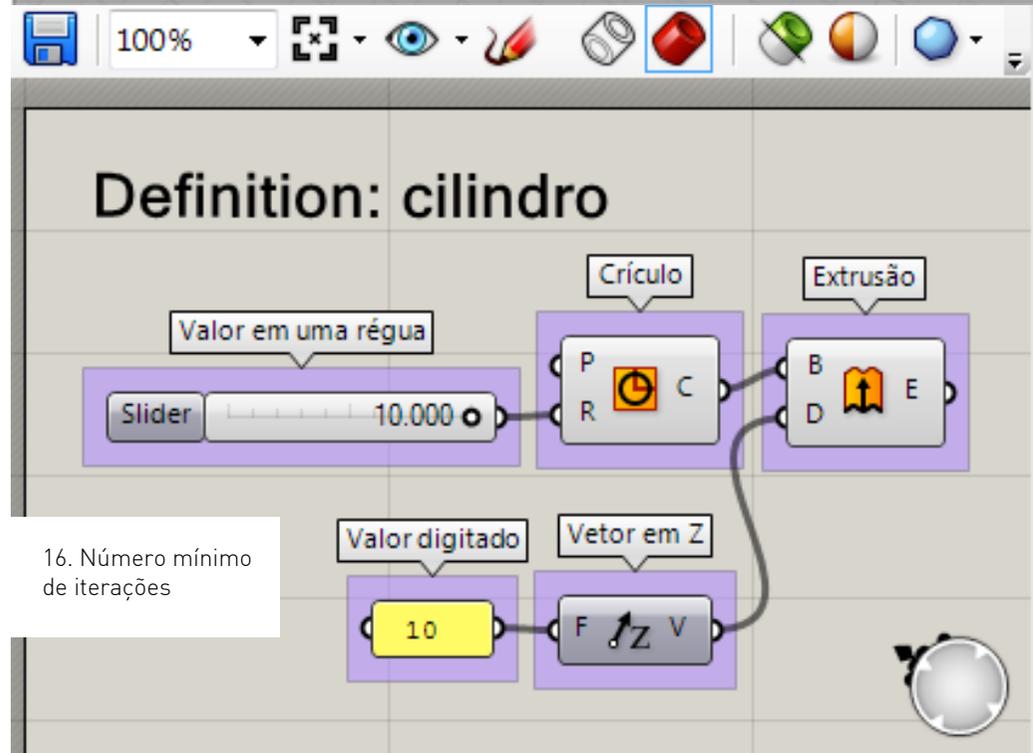
O programa se destaca por sua forma de trabalho e por sua interface gráfica. O usuário modela usando parâmetros, que são representados por peças com desenho muito semelhantes a componentes eletrônicos, a esses módulos são atrelados valores, que podem ser provenientes de outros parâmetros ou de ferramentas que guardam valores numéricos como régua ou prompts. Exemplo, para modelar um cilindro. Primeiro chamo o módulo que desenha círculos e ligo a letra R ( entrada para o raio do círculo) algum componente que forneça valores, no caso escolhemos usar um slider ( basicamente uma régua numerada) com o valor 10, isso cria um círculo de raio 10 milímetros.

Posteriormente passamos este desenho para o módulo extrude que gera o cilindro com 10 milímetros de altura no eixo Z. Este último valor é transmitido por uma caixa de texto. Esses valores podem ser modificados a qualquer momento e o modelo se modifica em tempo real, diferente da modelagem tradicional onde o trabalho deveria ser refeito.

Grasshopper é uma extensão bastante poderosa por tratar a modelagem como um script e por se tratar de uma ferramenta “open source” possui integração com muitas outras ferramentas. Através do “plug-in” firefly é possível transferir dados entre o programa e as plataformas arduino e processing. Este é um ponto muito importante para este projeto, uma vez que, podemos transferir as leituras de qualquer sensor para o grasshopper gerando novas formas e experiências de modelagem.



15. Modelo gerado parametricamente



16. Número mínimo de iterações

# Lista de componentes

## Potenciômetro

Componente eletrônico que possui resistência elétrica variável, sendo que o usuário consegue ajustar o valor girando uma chave. Todo potenciômetro possui três pinos, em um dos que estão nas pontas entra a corrente, o segundo é ligado no terra e o do centro é usado para ler o valor da resistência.

Existem potenciômetros em vários equipamentos de uso diário. Exemplo o controle de volume de caixas de som é feito por este tipo de componente, normalmente usando uma chave de 360 graus de giro. Já em “joysticks” de videogames encontramos potenciômetros de cerca de 90 graus de angulação.

## Botões

Também conhecidos como, chave momentânea. É um componente que interrompe ou permite a passagem de corrente. Nos modelos analógicos o circuito é interrompido fisicamente impedindo que uma extremidade do fio entre em contato com a outra, já os digitais apenas variam o valor de sua resistência e um controlador eletrônico reconhece essa informação e ativa ou desativa o sistema.

## Acelerômetro

É um sensor que percebe de maneira precisa a angulação e a velocidade dos movimentos realizados pelo objeto em que está instalado. Ele faz isso por meio de um mecanismo semelhante ao labirinto, uma das estruturas internas do ouvido responsável pelo equilíbrio, onde um líquido ou, conjunto de esferas presas por molas, movimentam-se livremente em um recipiente que capta suas posições.

## Sensores de infravermelho

Ou sensores de infravermelho passivos, são equipamentos que percebem a presença de tudo que emita raios infravermelhos com comprimento de onda entre 9 e 10 micrômetros, ou seja qualquer corpo que produza calor acima de 34 graus.

## Sensor de luminosidade

É um sensor que converte a luz em valor de resistência. Quanto maior é a intensidade luminosa menor é a resistência. Ele mede basicamente a incidência de raios ultravioleta.

## Sensor de temperatura

Os termo sensores do tipo NTC contêm óxidos metálicos tais como Cromo, Níquel, Cobre Ferro, Manganês e Titânio. Estes componentes diminuem a sua resistência elétrica com o aumento da temperatura.

## Piezo

É um sensor composto, normalmente, por um cristal de quartzo. Quando é usado de forma passiva emite um campo elétrico quando é submetido à pressão. Quando recebe uma carga elétrica externa ele se deforma, podendo ser usado para gerar sons.

## Protoboard

“é uma matriz de contatos para suporte a montagem de protótipos. Os sistemas experimentais em eletrônica, antes de serem montados em placas de circuito impresso e soldados, são testados por software e como protótipos em um protoboard.”

## Jumpers

Fios com ponteiros metálicas soldadas nas extremidades que fazem conexão entre os componentes eletrônicos, a protoboard e as placas de circuito.

# Algoritmos

## Fibonacci

É uma sequência numérica proposta pelo matemático Fibonacci no século XIII.

**(1,1,2,3,5,8,13,21,34,55,89,...)**

“Essa sequência tem uma lei de formação simples: cada elemento, a partir do terceiro, é obtido somando-se os dois anteriores. Veja:  $1+1=2$ ,  $2+1=3$ ,  $3+2=5$  e assim por diante.”

esta é considerada uma das maravilhas da matemática por representar formas perfeitas encontradas na natureza como é o caso de uma concha do mar.

## Desenhar com código (draw with code)

draw with code é um esquema lógico para criação de arte generativa bidimensional. O raciocínio é simples. Cria-se um elemento, este por sua vez é composta por uma forma geométrica e um conjunto de comportamentos.

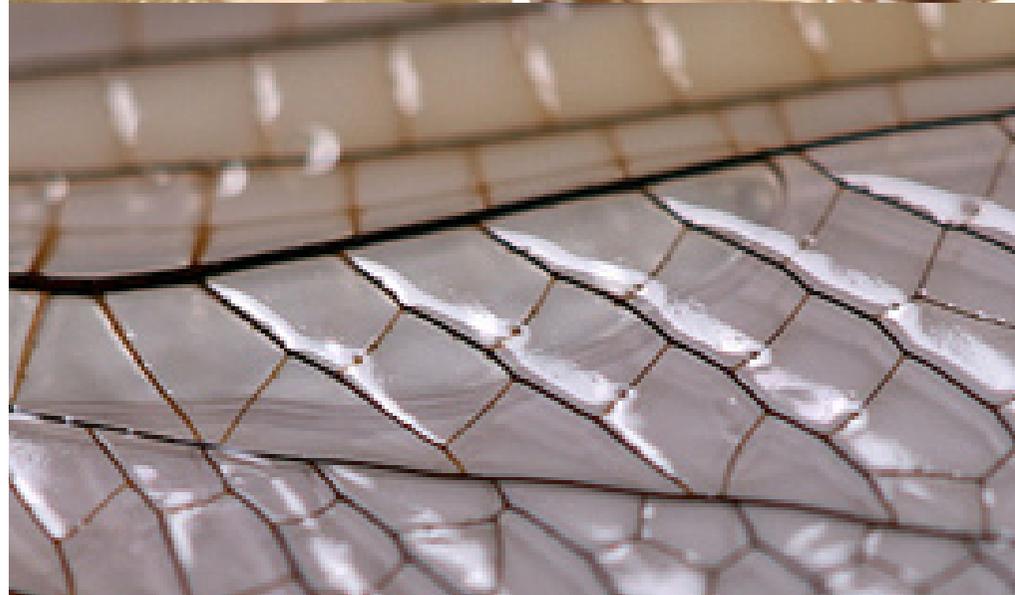
**Elemento = forma + comportamento 1 + comport 2 ...;**

Como exemplo de comportamento temos toda vez que um elemento colidir com outro é desenhada uma linha no de 100 pixels de comprimento do lugar da batida. Um dos aspectos mais interessantes deste algoritmo está na busca por um conjunto de elementos e ações que irão gerar um desenho harmonioso.

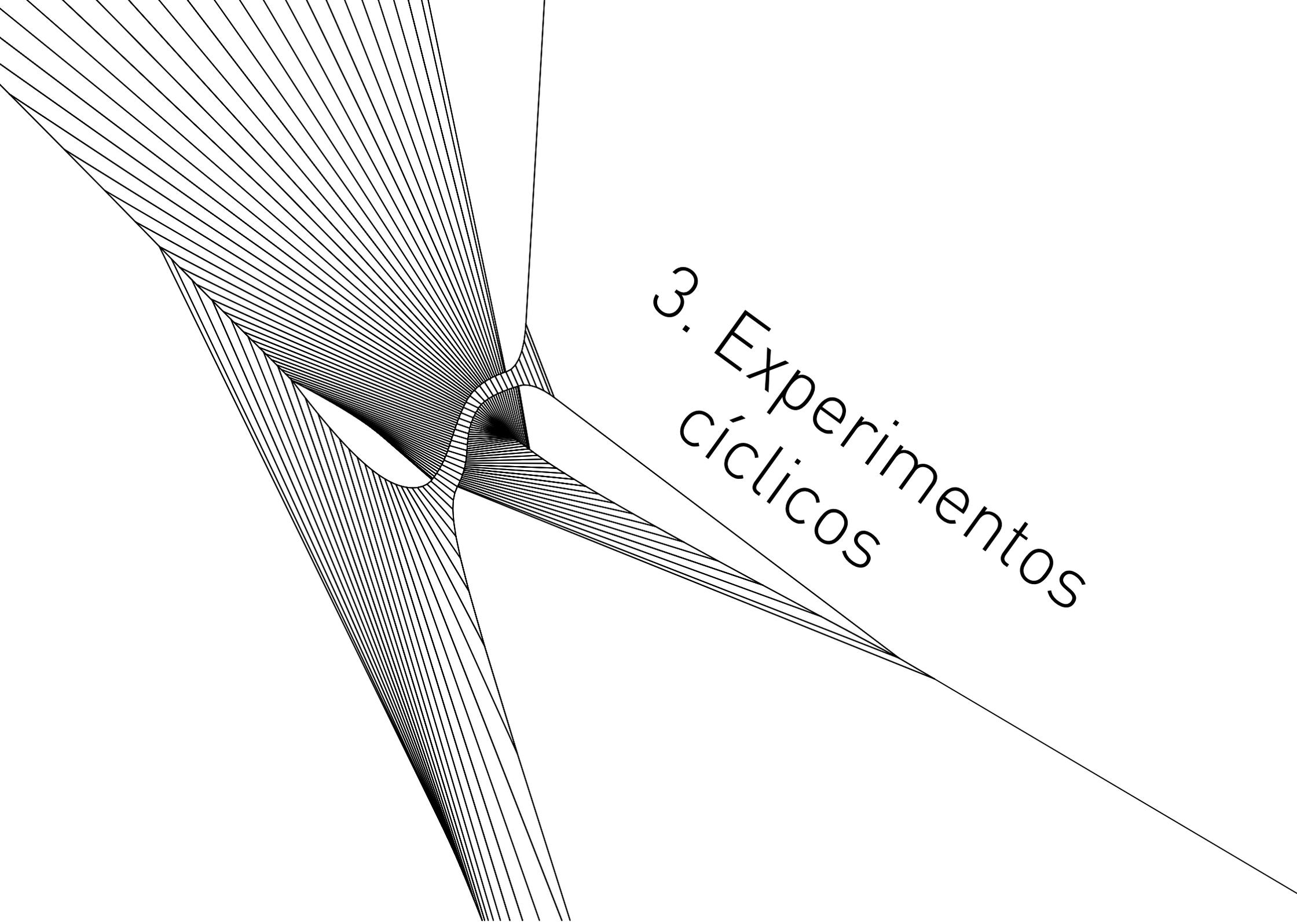
## Diagrama de Voronoi

O Voronoi é um tipo especial de diagrama matemático. Ele é formado a partir de pontos distribuídos em um dado espaço. A união das medianas, das distâncias entre cada ponto forma cada uma das células de voronoi.

É comumente presente na natureza. Em sua forma bidimensional podemos encontrá-lo no desenho de asas de insetos e folhas, e sua forma tridimensional é vista na espuma e em determinados fungos. Este diagrama pode ser aplicado em várias áreas, pois é uma decomposição do espaço tanto teórico quanto material, que possui grande resistência e é agradável visualmente devido a sua origem natural. Devido a isso todos os grafismos usados no projeto gráfico deste relatório são estruturas voronoicas bidimensionais.



17. Voronoi em suas formas



3. Experimentos  
cíclicos

## 3 Experimentos cíclicos

### 3.1 Primeiro experimento, controlador

#### 3.1.1 Proposta

Dei início a fase prática do projeto, propondo um experimento com o intuito de aprender como o processing trabalha com modelos 3D e a sua integração com arduino, já que não havíamos trabalhado com nenhuma dessas plataformas de desenvolvimento antes.

Para testar a comunicação entre eles eu idealizei um experimento que consiste na utilização de dois tipos diferentes de potenciômetro para interagir com o modelo de uma esfera na tela. Usei dois potenciômetros de 360 graus de giro, um controlando a iteração e o outro o tamanho da esfera. Em conjunto com estes utilizei um joystick, que é composto por dois potenciômetros de aproximadamente 90 graus de angulação, que controla a rotação da esfera no eixo x e y.

#### 3.1.2 Desenvolvimento

Para construir o experimento foram usados: uma placa arduino Duemilenove, uma protoboard, dois potenciômetros de 10 $\Omega$ , um “joystick”, dez jumpers um cabo USB.

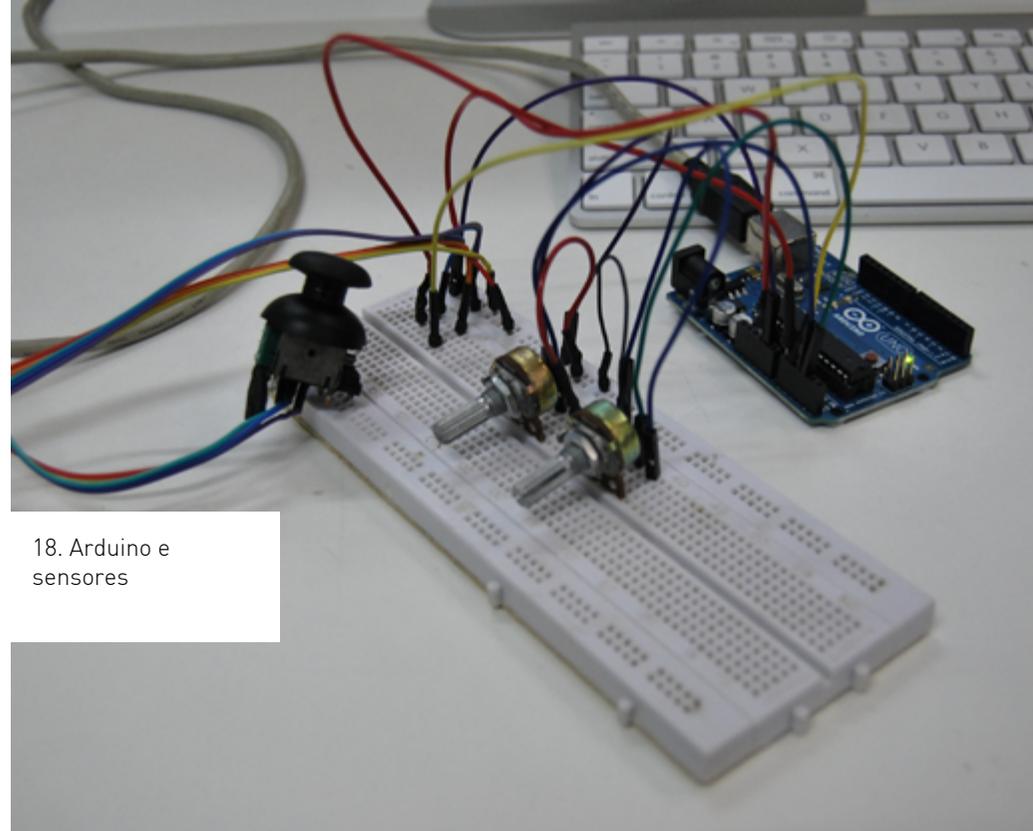
A protoboard foi usada para estruturar os componentes e possibilitar suas comunicações com a placa por meio dos jumpers. A leitura dos potenciômetros foi feita pelas quatro primeiras portas analógicas da placa. No lado do software, foi utilizada a biblioteca firmata, que torna possível que o processing controle as entradas e saídas da placa e que as leituras obtidas interajam com o que é mostrado na tela.

No decorrer do desenvolvimento da aplicação foram encontradas dificuldades importantes para compreender como os dados recebidos da placa devem ser tratados. Os potenciômetros de 10 $\Omega$  fornecem valores que vão de 0 até 360, logo que rodamos os primeiros códigos percebemos que não poderíamos jogar esses valores diretamente nas entradas de parâmetros, como o de escala. Quando o valor ultrapassa 100 a esfera fica tão grande que a câmera entra nela, para solucionar

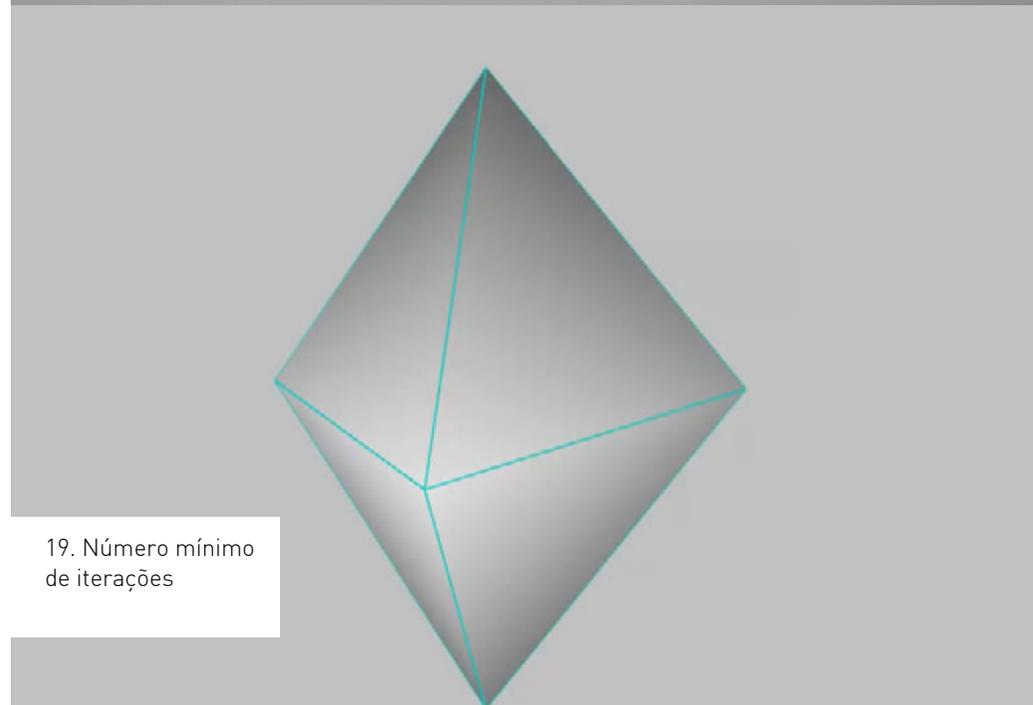
isso passamos a dividir os valores desta entrada por 4. Isso também foi feito para o parâmetro que controla as iterações, mas o valor da divisão foi 40. Já com relação às leituras dos potenciômetros do joystick se os valores fossem passados diretamente para os controladores da rotação a esfera pulava de uma posição para outra sem fazer uma rotação suave. Resolvemos este problema determinando que apenas os valores mínimos e máximos de leitura seriam usados. Exemplo quando o potenciômetro do eixo X está com valor zero a esfera gira no eixo X no sentido anti-horário e quando o valor é máximo ela gira no sentido horário.

### 3.1.3 Conclusões

O resultado do experimento foi satisfatório, pois cumpriu o objetivo de compreender como o processing trabalha com modelos tridimensionais básicos e percebemos que não devemos usar sem um tratamento adequado os dados provenientes das leituras dos sensores. Os gráficos obtidos nos fizeram refletir, quando o número de iterações estava no mínimo surgia uma pirâmide de faces triangulares chegamos à conclusão que



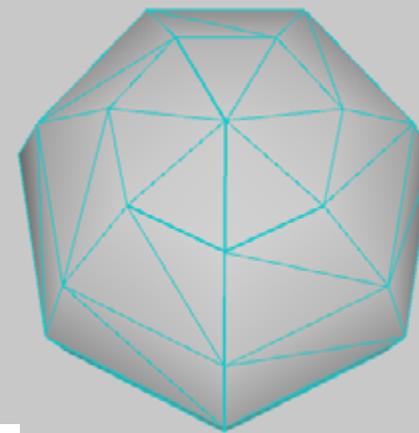
18. Arduino e sensores



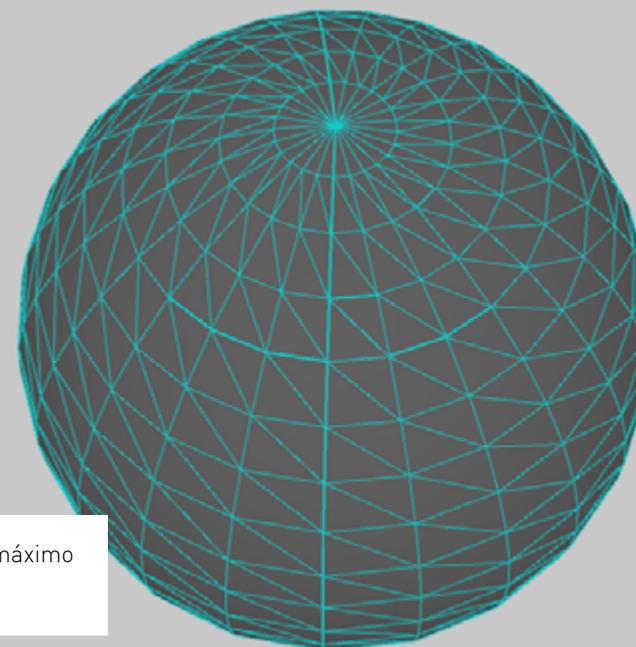
19. Número mínimo de iterações

esta é a representação mais básica de uma esfera por malha mesh. Uma conclusão insatisfatória foi a de perceber que toda a esfera que é criada com o comando nativo do processing vem com uma falha em sua malha representada cor um furo, até o momento desta publicação não foi encontrada uma solução para o problema.

No processing dentro de “void setup” podemos escolher duas bibliotecas de renderização o OpenGL e o P3D, sendo que a primeira joga todo o processamento gráfico para a placa de vídeo, não sendo uma boa opção para computadores antigos. Já a segunda usa o processador da máquina. Em nossos testes percebemos que o uso da biblioteca OpenGL gerou resultados mais fluidos que a P3D. Ressaltando que a máquina usada nos estudos era equipada com uma placa dedicada de um gigabyte de memória da Nvidia.



20. Número médio de iterações



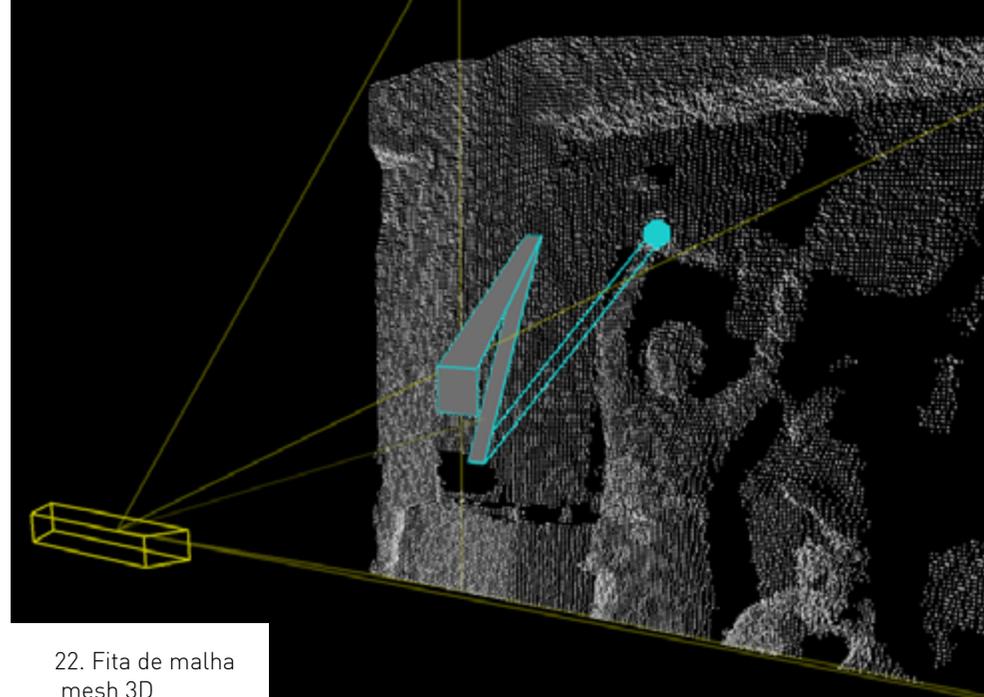
21. Número máximo de iterações

## 3.2 Segundo experimento

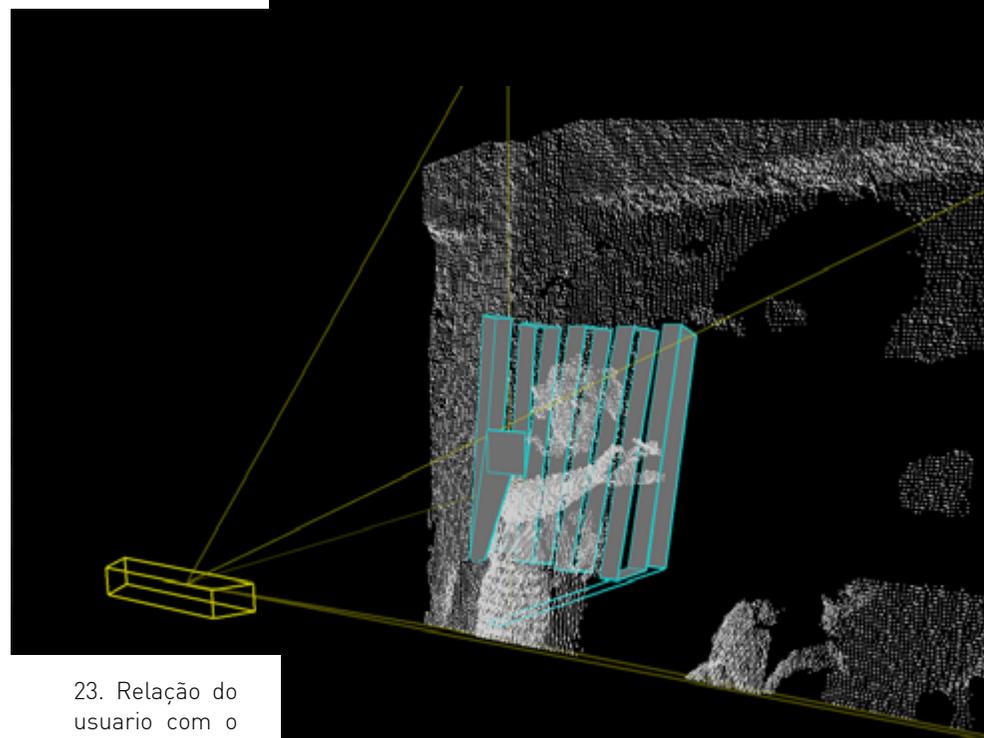
### 3.2.1 Proposta

Normalmente, quando fazemos uma modelagem em um software estabelecido no mercado, interagimos com ele por meio de ferramentas bidimensionais, o que frustra muitos usuários por não conseguirem passar para o computador o que desejam. Nesse experimento busco fazer o objeto ficar ao alcance do modelador.

Outro ponto é que, modelando da forma tradicional, perde-se a noção da escala do que estamos fazendo; propomos reverter essa situação usando como referência o próprio corpo do usuário. O experimento consiste na utilização de um Kinect para capturar os movimentos e o corpo do modelador. O usuário se vê na tela, representado por uma nuvem de pontos. Da mão trackeada pela Kinect surge um polígono, que acompanha seus movimentos até que o operador execute o click, deixando-o estático na posição desejada. Origina-se outro polígono de uma das laterais do primeiro, que pode ser movimentado da mesma forma e assim sucessivamente, formando uma fita de malha mesh.



22. Fita de malha mesh 3D



23. Relação do usuário com o modelo

### 3.2.2 Desenvolvimento

Em um primeiro momento procuramos basear o experimento em um código bastante simples que localiza o elemento mais próximo da câmera, porém, as leituras obtidas apresentavam muito ruído. Devido a isso tivemos que estudar o “track” de esqueleto, que é composto por uma sintaxe de código que, por ser orientado a objetos, é mais complexa. Por meio do livro “make things see” tivemos acesso a vários códigos de rastreamento do corpo, todos comentados em detalhes. Em um dos exemplos descobrimos que, por meio do comando `handVec.x`, `y` ou `z` nós obtínhamos as coordenadas de uma das mãos do usuário diretamente da biblioteca `simple open NI`.

Uma vez resolvido o problema do “track”, precisávamos desenvolver o “motor 3D,” que geraria as faces da fita. Diferente do exercício anterior este usa um modelo complexo que não é formado por primitivas como cubos e esferas. No `processing` basta escrever as coordenadas dos vértices dos polígonos dentro de um comando específico que ele passa a reconhecer estes como pertencentes a um único objeto. Foi preciso uma

pesquisa para encontrar uma função de incremento de array do `processing` que possibilitava a criação de modelos com um número indeterminado de lados.

### 3.2.3 Conclusões

Todo modelo que é criado nesse experimento obedece a escala do mundo real e o posicionamento também. Ou seja, se uma parede é desenhada nessa aplicação e ela está na minha frente, eu não me verei mais, porém eu posso passar por ela como um fantasma. Fazer esse tipo de “brincadeira” mostra uma nova relação com a modelagem, uma vez que nos programas tradicionais estamos presos do outro lado da tela sem uma representação nossa virtual. Esse experimento possui pendências. Pretendemos abolir a necessidade do uso de hardwares para executar o click, para isso estamos nos aprofundando nos estudos da biblioteca `simple open NI` e no reconhecimento de esqueleto.

## 3.3 Terceiro experimento

### Modelagem com primitivas

#### 3.3.1 Proposta

Fazer rápidos sketches no espaço 3D usando primitivas tridimensionais básicas como cubos e esferas. Distribuir elementos virtuais em uma cena, usando os “inputs” de dados tradicionais é uma tarefa complexa principalmente para os iniciantes, por dois motivos: Primeiro, para criar uma primitiva é preciso conhecer a interface do programa em questão. Segundo, o usuário não pode pegar o objeto e colocá-lo onde deseja no espaço, ele precisa dividir sua atenção em três ou mais telas distintas que representam as vistas da cena.

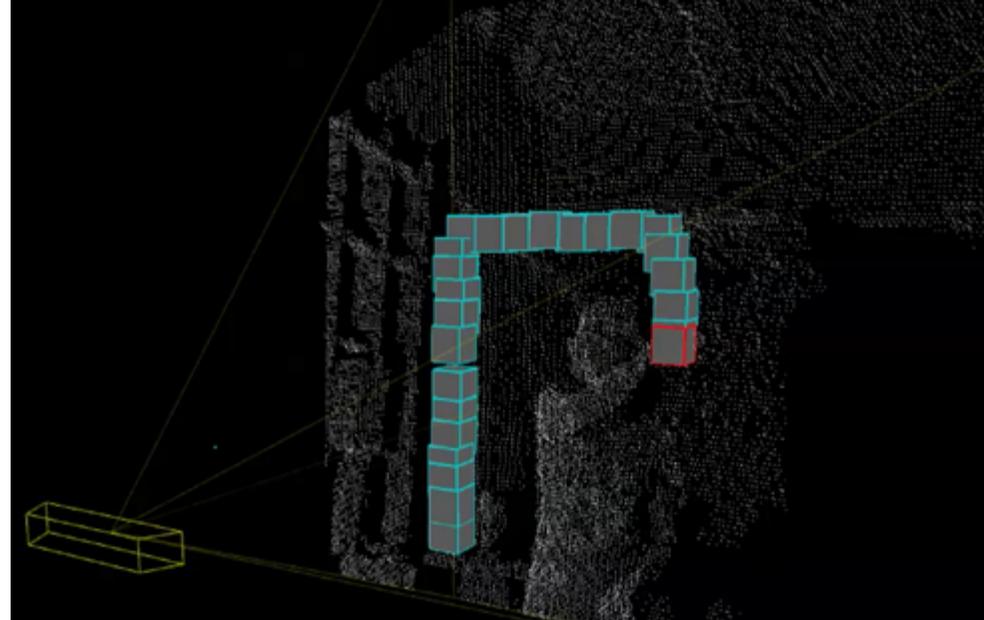
Quando o usuário se posiciona na frente da instalação, um cubo, de arestas vermelhas, surge em uma de suas mãos, e passa a seguir seus movimentos. Até que este execute um clique, no mouse, travando a posição do cubo no espaço e ele passa a ter um contorno azul claro. Instantaneamente outro cubo surge na mão do modelador, que pode repetir o processo.

Foi construído um desdobramento deste experimento. No momento em que o click é executado e o cubo torna-se estático o usuário pode mover a mão para frente e para trás modificando a escala do objeto; quanto mais distante do ponto de fixação maior é a escala. Quando o clique é executado novamente a escala torna-se fixa e o objeto passa a rotacionar. Se a mão do modelador se move no eixo X o objeto gira em torno desse mesmo eixo, e a mesma coisa ocorre para as coordenadas Y e Z. Quando o último clique é feito o objeto fica travado e o processo recomeça.

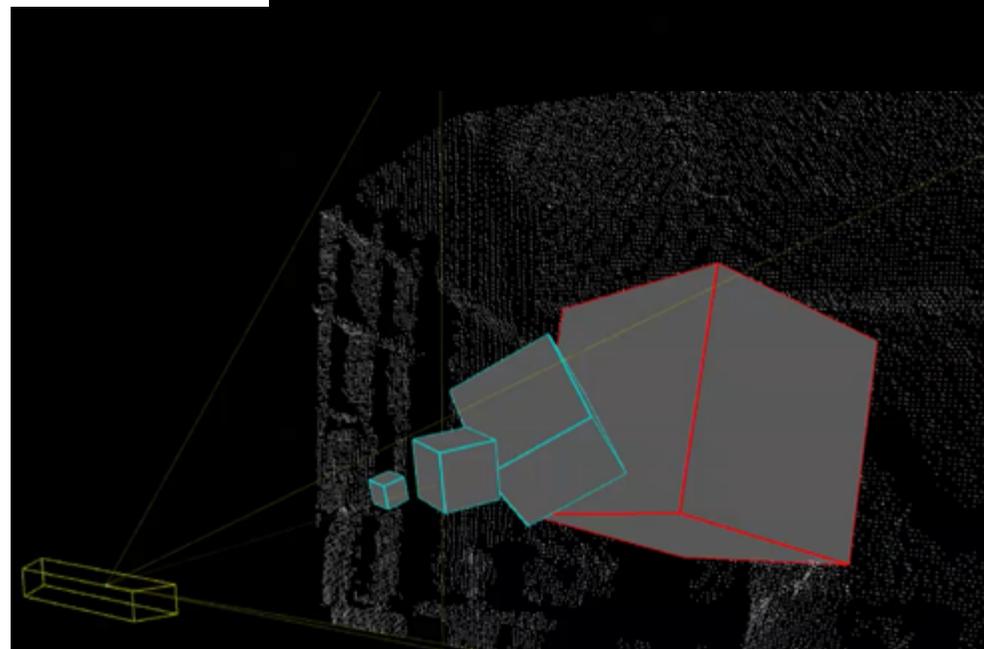
#### 3.3.2 Desenvolvimento

Graças ao experimento anterior este se tornou bastante simples. Escrevemos um programa que localiza a posição de uma das mãos do usuário e desenha sobre ela uma primitiva geométrica pré determinada. O programa percebe quando um clique é executado e armazena as coordenadas da mão neste momento, em uma lista. Por fim o programa desenha todas as primitivas da lista na tela. Esse processo ocorre para cada frame que é exibido.

Amaiordificuldadefoinodesdobramento do experimento. Para criar a função de escala e rotação fizemos um código que, percebe a distancia entre o ponto de fixação do objeto e a mão do modelador. O valor obtido é passado diretamente para o objeto, determinando seu tamanho. Exemplo: se a mão do usuário está a 10 cm da posição fixa do cubo, este terá 10cm de aresta. No caso da rotação é diferente, captamos a distancia em eixos especificos. Exemplo se a mão se distancia 10cm no eixo X esse valor é diretamente convertido em radianos e passado como valor de rotação no eixo X o mesmo ocorre para o eixo Y. A tarefa de construir este código tornou-se menos complexa graças a uma função própria do processing chamada “dist” que mede a distancia entre dois pontos; em outras linguagens é preciso criar uma função trigonométrica especifica.



24. cubos de tamanho fixo



25. cubos com escala e rotação

### 3.3.3 Teste com usuários

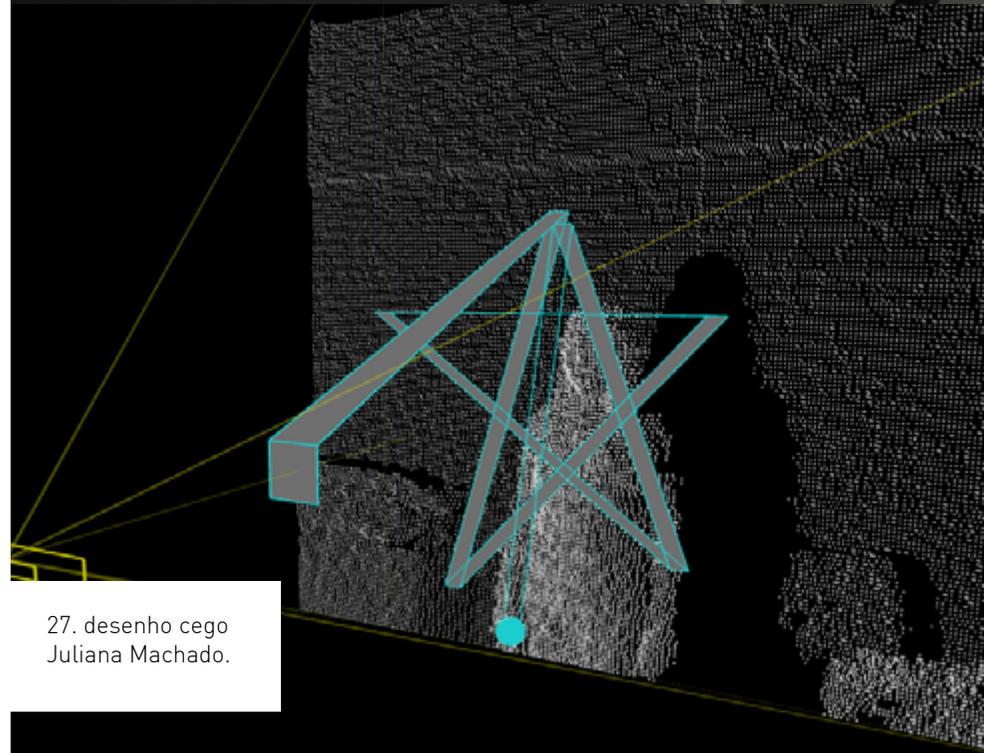
A primeira leva de testes serviu como preparação para a seguinte do projeto, que será focada na usabilidade, para perceber na prática como funcionam o segundo e terceiro experimentos. A dinâmica sucedeu-se da seguinte forma: Cada usuário testou os três experimentos construídos com Kinect de maneira livre, apenas para perceber como eles funcionam. Posteriormente foi pedido para que cada um escolhesse a proposta que mais gostou e modelasse alguma coisa olhando para tela do computador. Em seguida a tela foi retirada, e o usuário tentou modelar a mesma coisa usando apenas a sua localização espacial como referência.

Durante a execução dos testes anotamos os relatos dos usuários e criamos uma lista com as considerações mais pertinentes:

Foi unânime a confusão em perceber a profundidade quando a câmera era posta de frente, semelhante a um espelho. “Não sei bem se estou em um ponto mais próximo ou mais distante” disse o usuário Marcello Cortez. Para ele uma possível solução seria tornar maior e mais claro o objeto quando estivesse próximo da câmera, e o oposto aconteceria conforme a distância aumentasse.



26. Montagem do teste.



27. desenho cego  
Juliana Machado.

## Din3D - Experimentos cíclicos

Com a câmera em 3/4, a profundidade é mais nítida, mas o recurso citado anteriormente pode reforçar esta percepção.

Questionaram se a visualização do objeto estava correta. Por exemplo: quando um modelo ocupa toda a tela, ele acaba cobrindo a representação do usuário. Uma proposta foi reduzir a opacidade e valorizar os contornos.

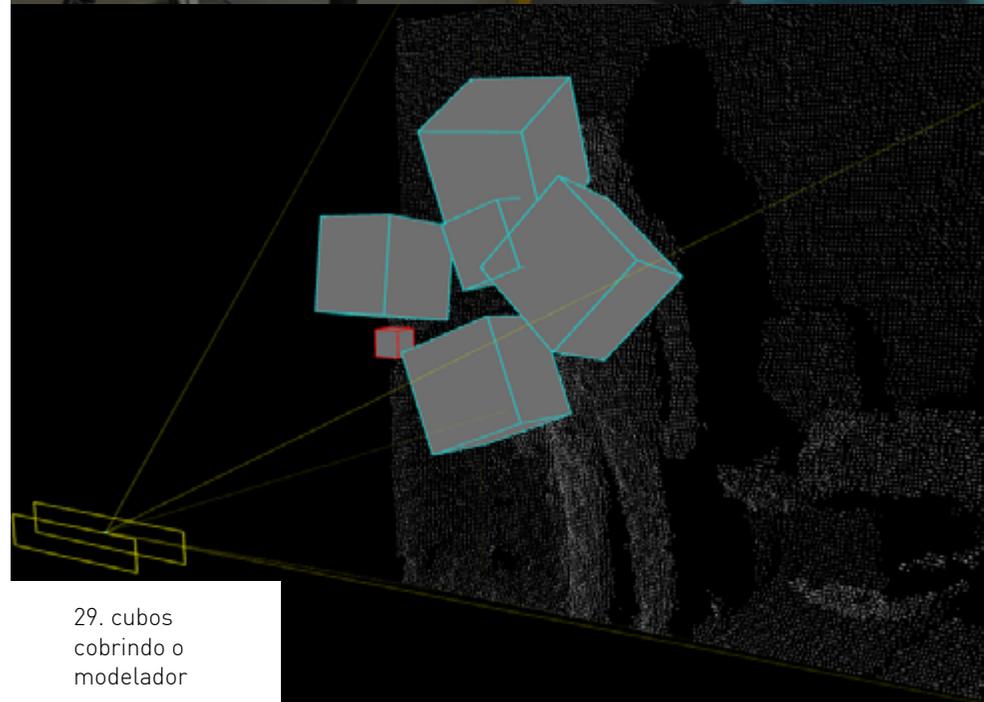
Outro dado unânime foi a preferência por modelar olhando para a tela, onde os resultados eram mais próximos dos desejados. Juliana Machado disse que “Olhando para a tela você é mais preciso, mas sem olhar você é mais livre.” Consideramos que a perda de uma referencia visual gera um certo desconforto ao usuário.

No experimento das fitas foi relatada a limitação de não poder escolher a aresta de base para a próxima face e de não ser possível unir uma face na outra. Isso impede a formação de um objeto fechado. Sentiu-se a necessidade de poder usar as duas mãos para modelar.

Na modelagem de primitivas com parâmetros todos consideraram intuitiva a forma de aumentar e reduzir o modelo, já no momento de rotacionar consideraram o movimento brusco. Como solução sugeriram tornar a captação dos movimentos menos “sensível”.



28. Juliana Machado, modelando



29. cubos cobrindo o modelador

Os entrevistados que já têm algum conhecimento em modelagem, sentiram falta de poder girar a câmera enquanto desenhavam . Sugeriram poder modelar com uma mão e girar com a outra.

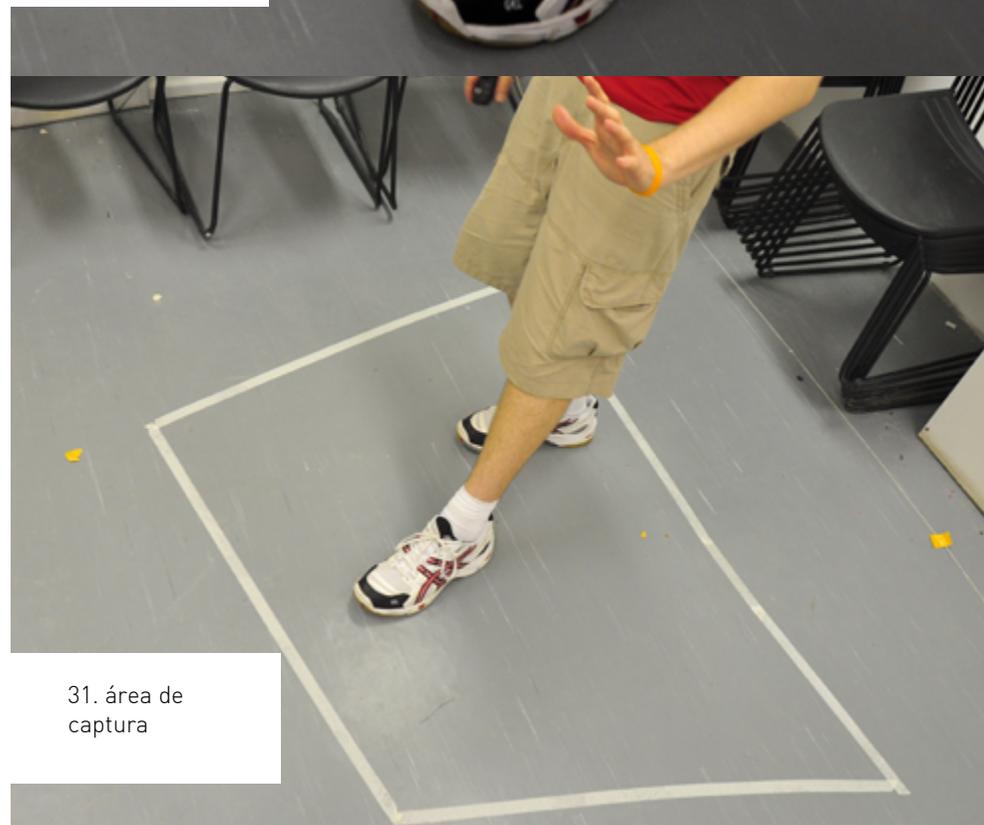
No experimento de primitivas, um dos usuários falou da possibilidade de fazer uma série de objetos apenas segurando o botão do mouse semelhante a um brush em softwares de desenho.

### 3.3.4 Considerações

Este teste foi o primeiro contato do trabalho com o público e levantou questões pertinentes para o futuro do trabalho. Nos primeiros testes os usuários que entraram na sala se depararam com um notebook em cima da mesa e com um Kinect sobre um tripé a sua frente, a única referencia espacial que existia era uma fita colada no chão representando a distancia recomendada de captura ( cerca de 1,80m ). Duas pessoas fizeram desta forma o teste e obtivemos pouca exploração do eixo Z, o resultado foram desenhos com a lógica bidimensional e moviam-se pouco apenas levando a mão de um lado para o outro. Para contornar esse problema, sugeriram marcar no chão toda a área de captura. O resultado foi uma melhor percepção da profundidade e aproveitamento de toda a área de trabalho. A delimitação da área de captura causou uma sensação de liberdade para o modelador se movimentar.



30. marca de distância mínima

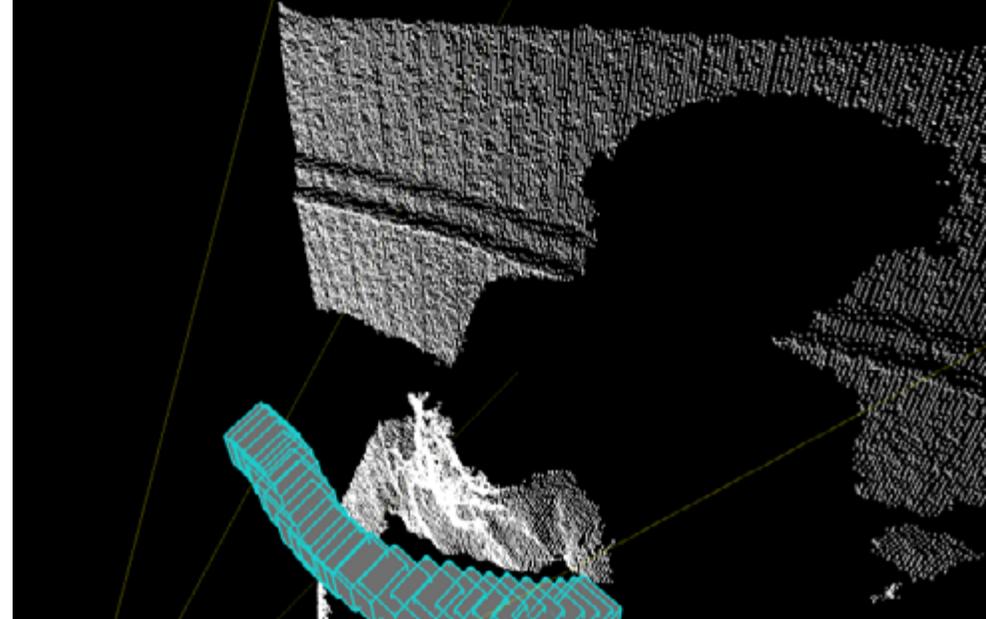


31. área de captura

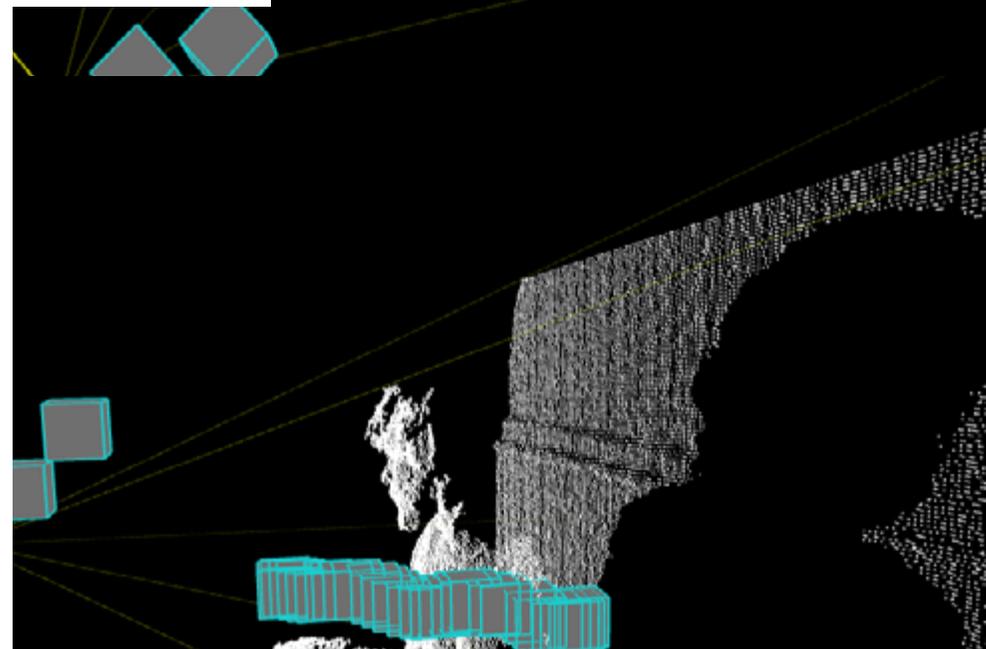
Neste processo nos deparamos com um “ruído,” o qual batizamos de efeito compasso. Experimente ficar parado e fazer uma linha reta da direita para a esquerda, quando olhamos essa reta de frente ela parece correta, mas se a virmos de cima percebemos que ela é torta no eixo z; isso acontece pois a mão não flutua no espaço, ela está fixa em uma estrutura onde o ponto de apoio é o ombro, gerando uma leve curvatura no movimento. Como o kinct é muito sensível à profundidade, ele capta tal ruído. Depois, com a delimitação da área de captura, surgiu outro padrão. Agora os usuários mantiveram a mão parada e andaram para fazer a linha. Esta parecerá reta se vista de cima para baixo, mas nas demais posições percebemos um desenho em forma de onda causado pelo ato de caminhar.

Perceber esse tipo de comportamento é crucial para um bom design de interação, pois podemos assimilar isso no trabalho ou buscar formas de eliminar este ruído. O nome que demos remete ao fato que apenas uma pessoa com as habilidades de mímico conseguiria fazer uma linha totalmente reta.

O projeto até este ponto é considerado “alfa,” o que na prática representa um software que tem muito o que evoluir, até porque muitas funções ainda não foram incorporadas. Por meio deste teste buscamos perceber de quais funções os usuários sentiram falta, para serem implementadas nos próximos experimentos:



32. Efeito compasso no eixo Z.



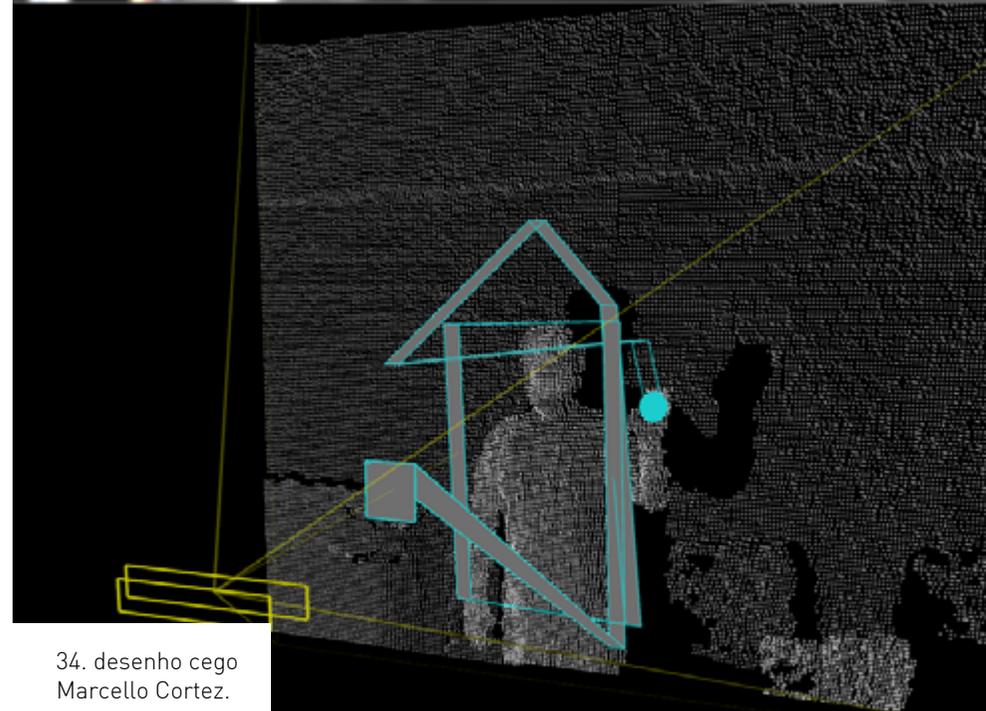
33. Efeito compasso no eixo Y.

Modeladores com experiência sentiram falta de um comando comum no mercado, o “snap”, principalmente no experimento fita. Com este comando o usuário pode fixar o ultimo quadrado criado em uma das extremidades do primeiro, para formar um objeto fechado.

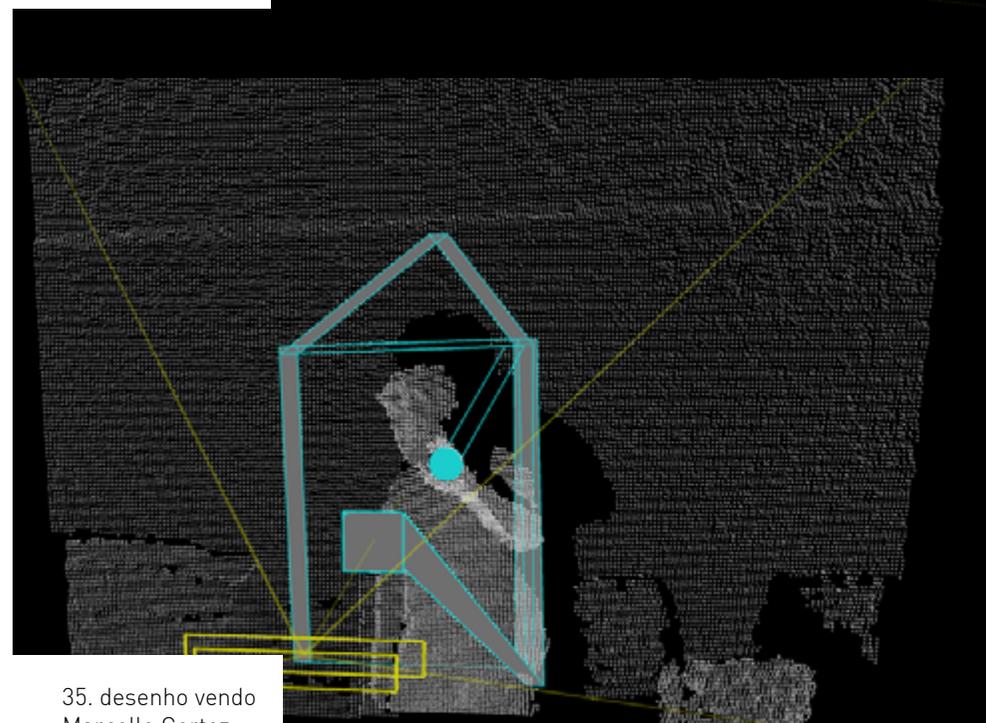
O Kinect captura a imagem 3D da seguinte forma: Imagine uma imagem 2D convencional; cada pixel carrega uma informação de cor e de localização nos eixos X e Y que são dadas pela ordem em que este aparece na tela. Já o Kinect acrescenta a posição Z em cada pixel, gerando o que chamamos, no mundo da modelagem virtual, de nuvem de pontos, onde cada pixel representa um ponto no espaço. O usuário é representado na tela como um objeto 3D e, graças a isso, podemos criar uma câmera virtual para capturar a imagem do modelo do usuário junto com o objeto modelado.

Os usuários ficaram impressionados por aparecerem na tela, só que quando a câmera virtual foi movimentada ficaram momentaneamente confusos. “ se o Kinect está na minha frente como eu estou sendo visto de cima? “ perguntavam e olhavam para a direção onde a câmera virtual deveria estar, sem encontrar nada.

Ressaltamos como grande ponto positivo do projeto e ,que o difere de outros programas conhecidos, está na simplicidade e na velocidade com que é possível criar um objeto 3D.



34. desenho cego  
Marcello Cortez.



35. desenho vendo  
Marcello Cortez.

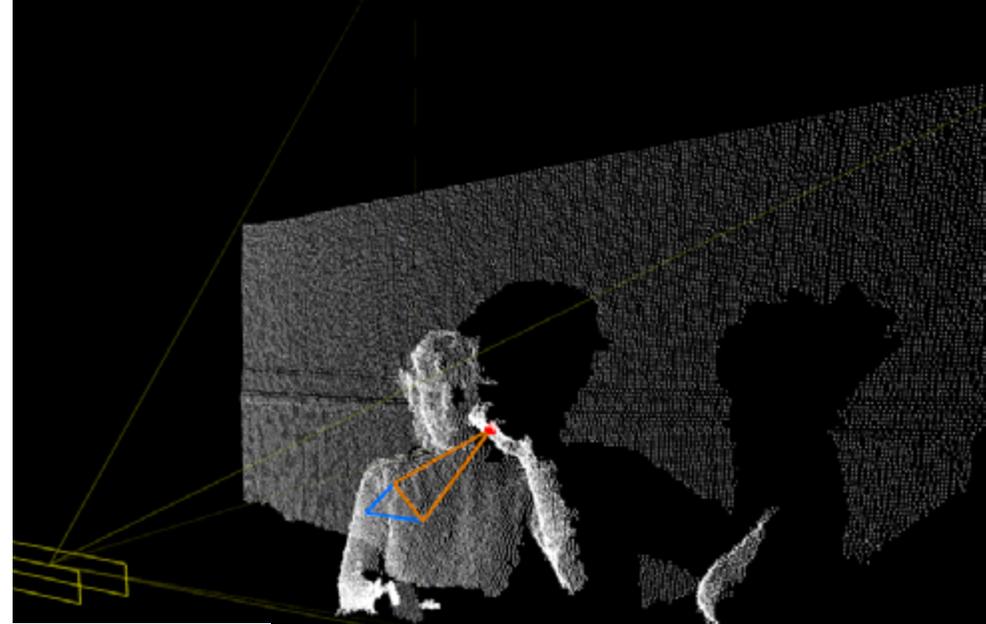
## 3.4 Quarto experimento Modelagem poligonal triangular

### 3.4.1 Proposta

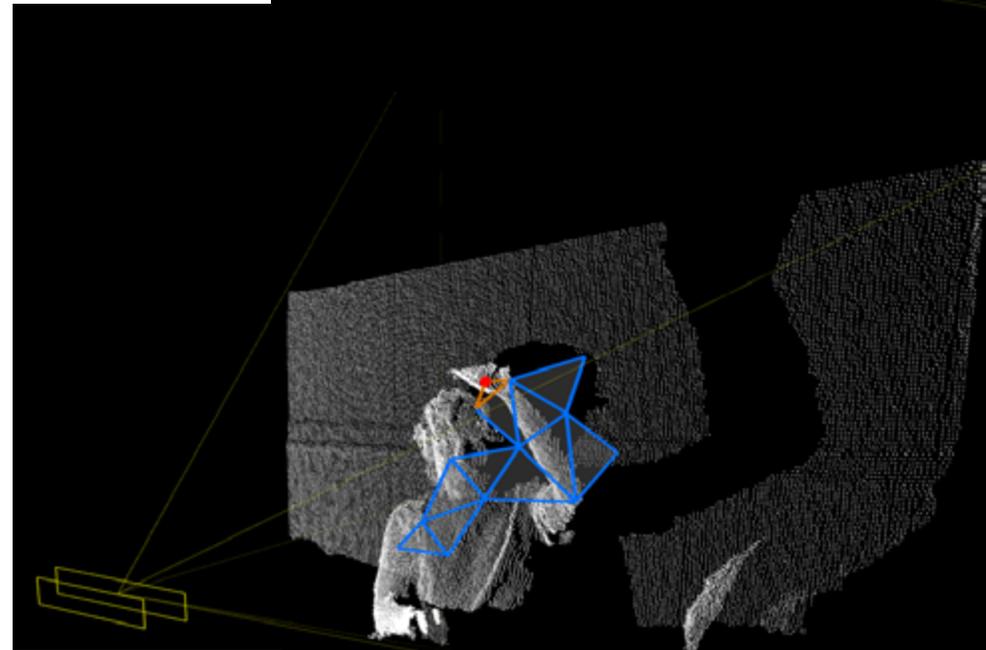
Este experimento é um desdobramento do “Fita,” levando em conta os dados coletados no teste de usabilidade.

Quando um usuário se posiciona na área de captura, surge um triângulo preso em sua mão. A posição do objeto é travada com um click. Em seguida partem duas linhas para a mão do usuário, vindas do vértice mais próximo da localização da mão dele. Chamaremos estas retas de linhas de construção e os vértices de ancoras. Se o operador mover a mão para outro ponto as linhas mudam seu vértice de fixação, buscando as menores distancias. Quando um click é executado este novo triângulo fica estático e o processo recomeça.

Se em algum momento surgir um buraco na malha em construção, basta que o usuário posicione sua mão no local do defeito e precione a tecla “S” para ativar a função “snap”. Com esta ferramenta as linhas de construção ficam ancoradas nos vértices desejados e, com um click, é construído um triângulo de reparo.



36. Primeiro triângulo.

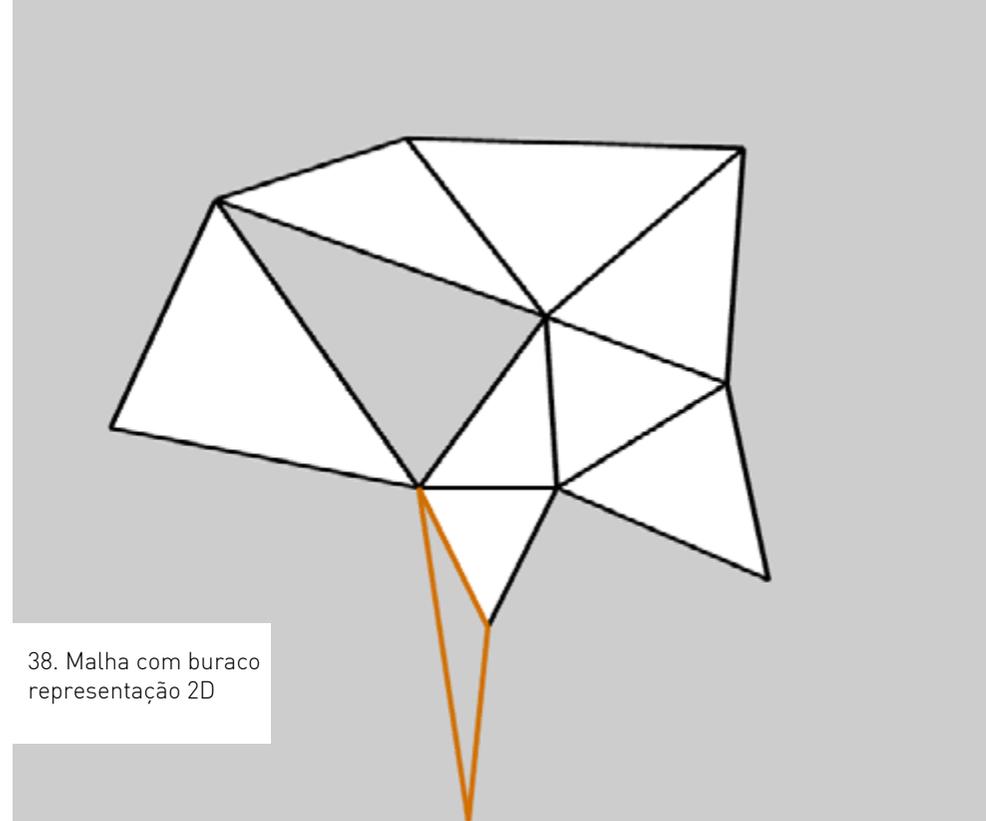


37. fita de malha triangular.

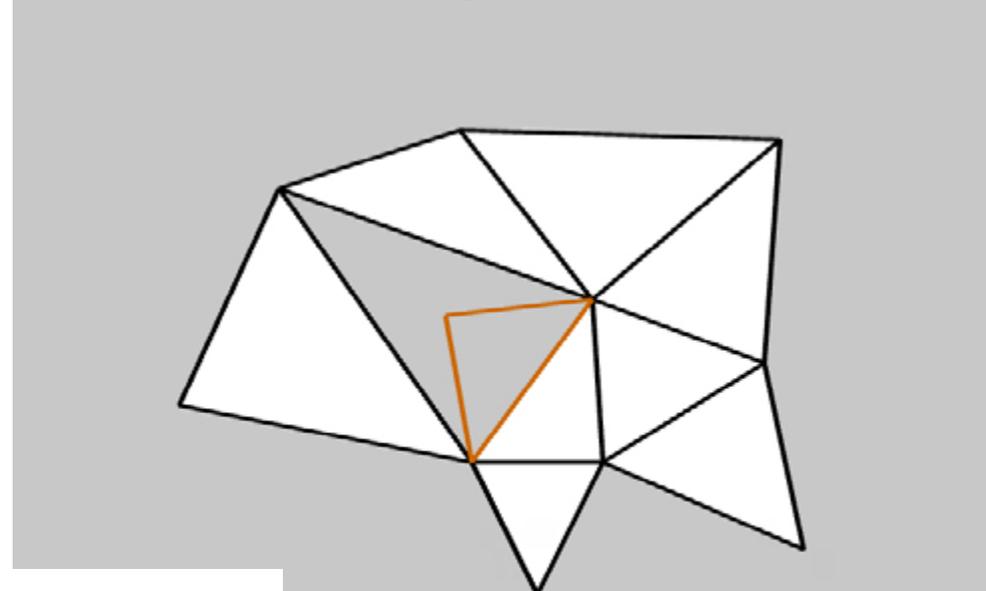
### 3.4.2 Desenvolvimento

Como pode ser percebido, cada experimento aborda os conhecimentos adquiridos com o anterior e acrescenta novos elementos em uma metodologia segundo a qual o projeto evolui em pequenos passos. No desdobramento do experimento com primitivas aprendemos a medir a distancia entre um ponto e a mão do usuário. Partindo deste princípio escrevemos um código que localiza os dois vértices mais próximos da mão do usuário e desenha linhas entre ambos. Na função “snap” os últimos vértices âncoras ficam travados e o terceiro ponto deixa de ser a mão do usuário para o vértice mais próximo desta. Quando o clique é executado, os pontos são guardados em uma lista que armazena todas as faces do objeto. Antes da implementação com a câmera 3D fizemos um programa para testar a aplicação sem o eixo Z.

Optamos por trabalhar com faces triangulares pois são matematicamente mais simples de lidar e, quando um modelo é exportado para outros programas, todos os quadrados são divididos em triângulos.



38. Malha com buraco  
representação 2D

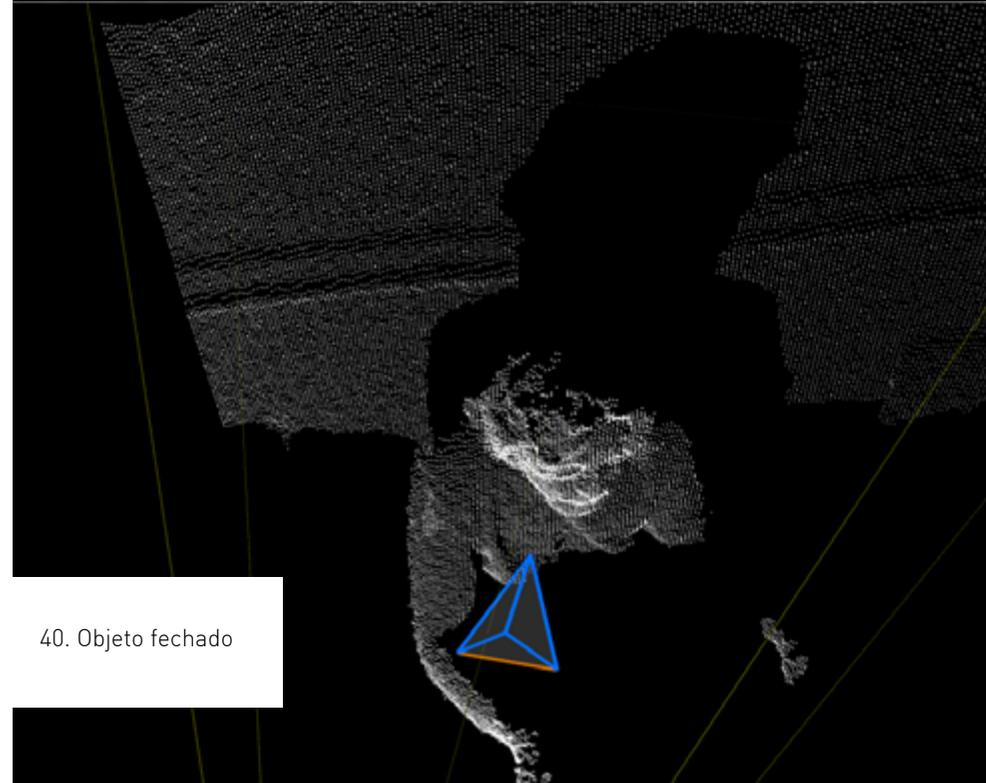
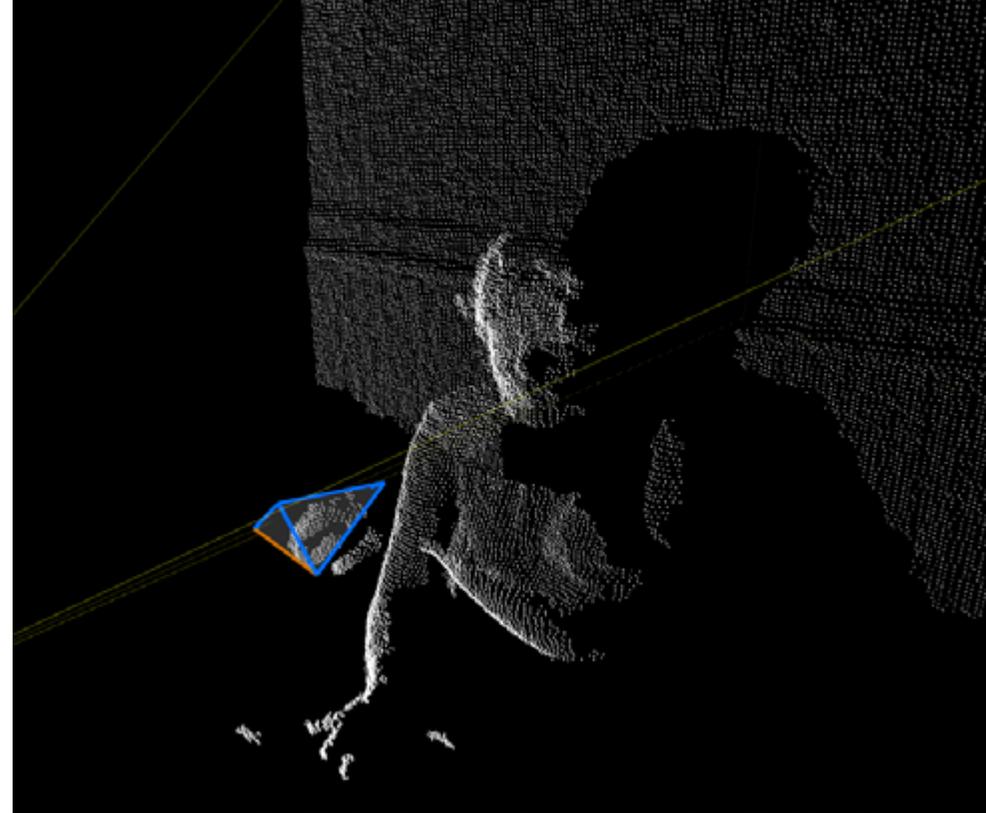


39. Reparando a  
malha com “snap”

### 3.4.3 Conclusão

Essa é uma ferramenta que pode construir modelos virtuais complexos baseados em polígonos, de uma maneira bastante próxima da modelagem tradicional. Para chegar neste resultado tivemos que incorporar comandos de teclado, o que torna a aplicação menos intuitiva para iniciantes. O experimento, portanto, é voltado para profissionais e pessoas com boa orientação espacial.

Diferentemente do estudo “fita”, com este é possível criar um objeto fechado pronto para ser exportado para outros softwares ou protótipado em uma impressora 3D.



40. Objeto fechado

## 3.5 Quinto experimento Escultor

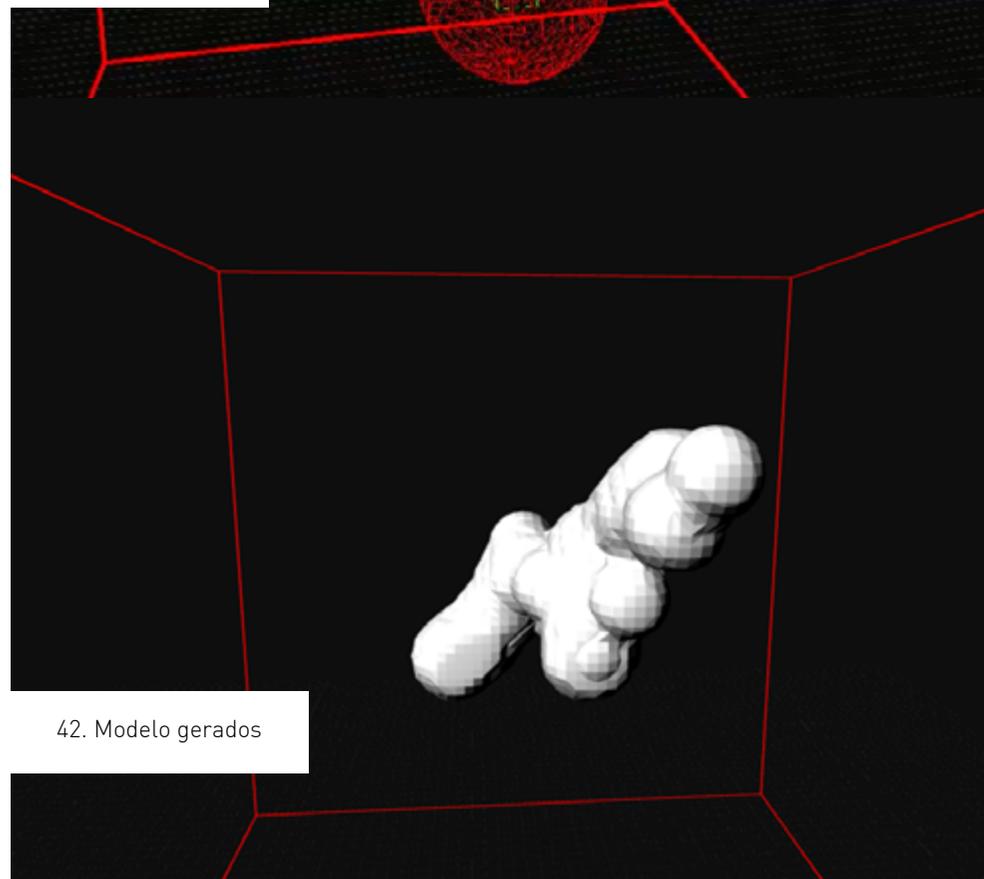
### 3.5.1 Proposta

Este experimento não estava previsto, mas como ficamos adiantados em relação ao cronograma o fizemos como alternativa ao anterior. Este busca gerar um modelo complexo da forma mais simples possível.

Neste o usuário se posiciona na frente de uma tela que exibe um cubo representado apenas por linhas, no centro do cubo existe uma esfera, que começa a se mexer de acordo com uma das mãos do operador. Diferentemente dos experimentos anteriores, este é em primeira pessoa; o usuário não vê nenhuma representação sua, devido a função “head tracker”, que percebe a posição da cabeça do operador e posiciona a câmera sobre ela, aumentando a imersão do usuário. Quando o manipulador clica no mouse, a esfera começa a gerar uma malha formada por polígonos triangulares, com um aspecto de argila, até que este clique novamente. O cubo representa os limites da área de trabalho, fora dele nenhum desenho é executado.



41. cursor 3D



42. Modelo gerados

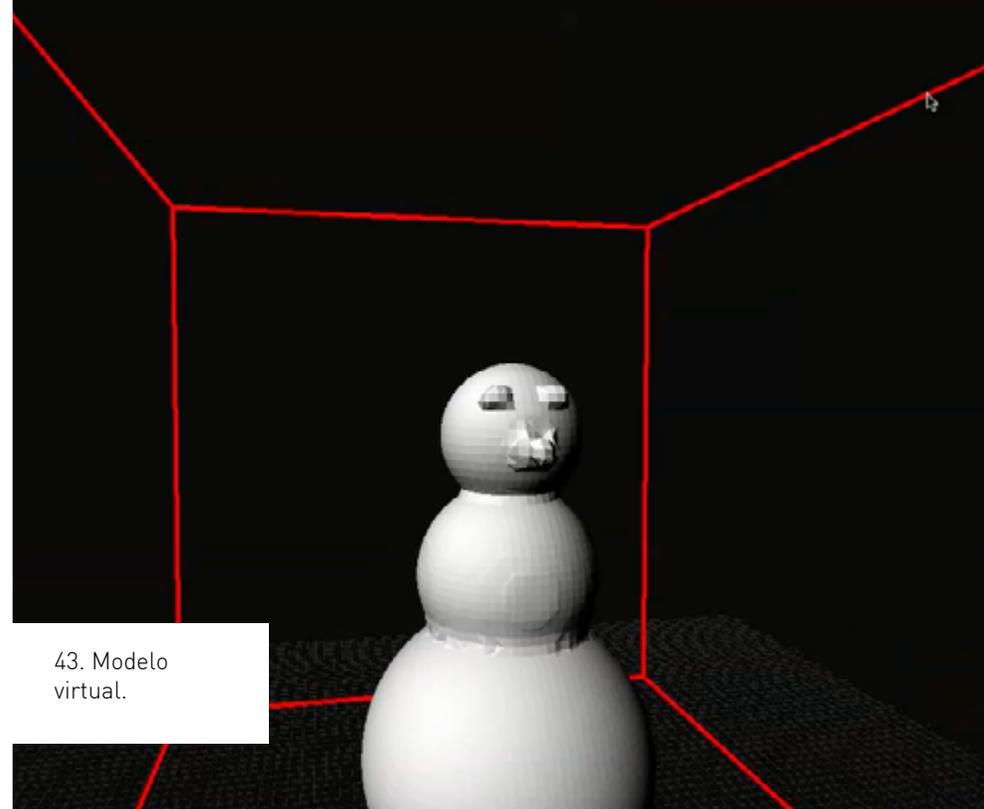
### 3.5.2 Desenvolvimento

Este foi o experimento mais complexo de ser executado, uma vez que, nos anteriores não usamos nenhum recurso externo ao processing para gerar os modelos . Neste caso usamos uma biblioteca chamada “Toxiclibs”. Outro ponto de dificuldade foi a integração entre os dados recebidos do kinect e a maneira com que esta biblioteca os processa. Toxiclibs gera uma área de trabalho tridimensional diferente da criada pelo Kinect, exigindo uma calibragem para sincronizar os dois espaços.

Para a função “head tracker” foi criado um algoritmo que descobre a posição da cabeça do usuário e coloca a câmera sobre ela. O target da câmera foi setado para o centro do cubo, para que o modelo não saia do campo de visão.

### 3.5.3 Conclusões preliminares

Como nesta aplicação o modelador não precisa se preocupar em fechar formas como no anterior ele possibilita que um usuário comum crie um modelos para prototipagem em impressoras 3D



43. Modelo virtual.



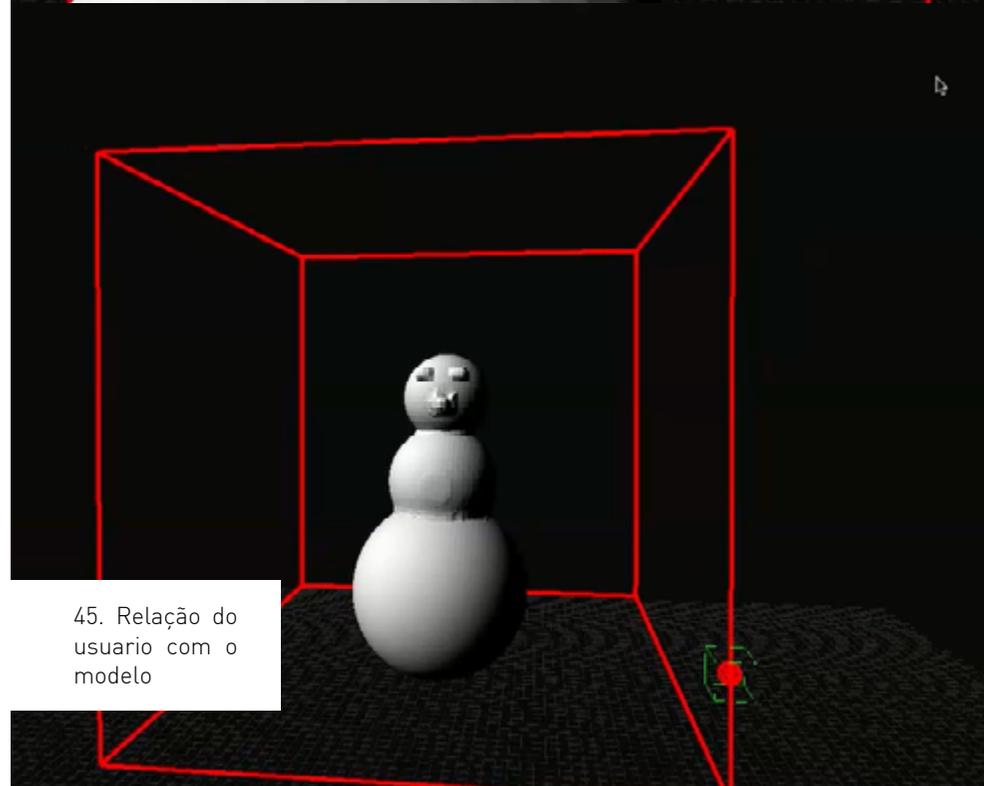
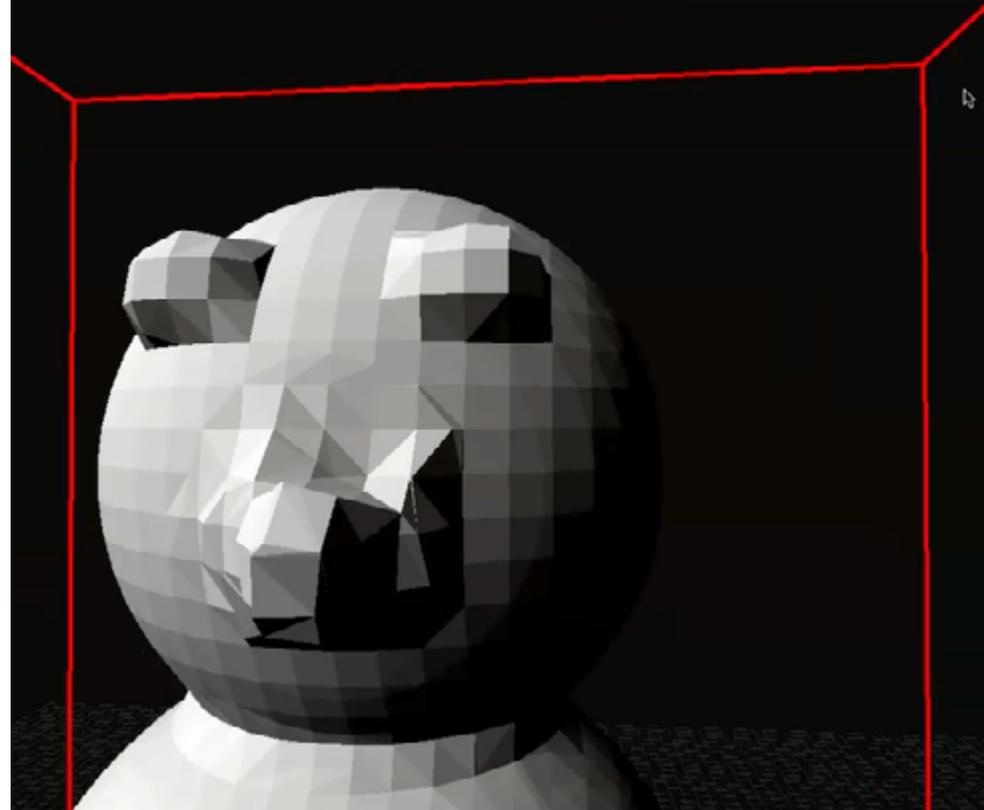
44. Modelo prototipado

O “head tracker” foi uma função desenvolvida grassas ao relato coletado nos testes de usabilidade que mencionava a falta de uma forma mais intuitiva de visualizar o objeto em outros ângulos. Em avaliações preliminares esse recurso se mostrou eficiente, por ser simples e confortável de usar. Surgiram situações inesperadas e positivas como uma maior exploração de todo o espaço para modelagem e a utilização da função como zoom, bastando alguns paços para frente.

### 3.6 Escolha de um caminho e Passos futuros

Como fechamento do projeto será produzida uma instalação interativa que utiliza um dos experimentos como base.

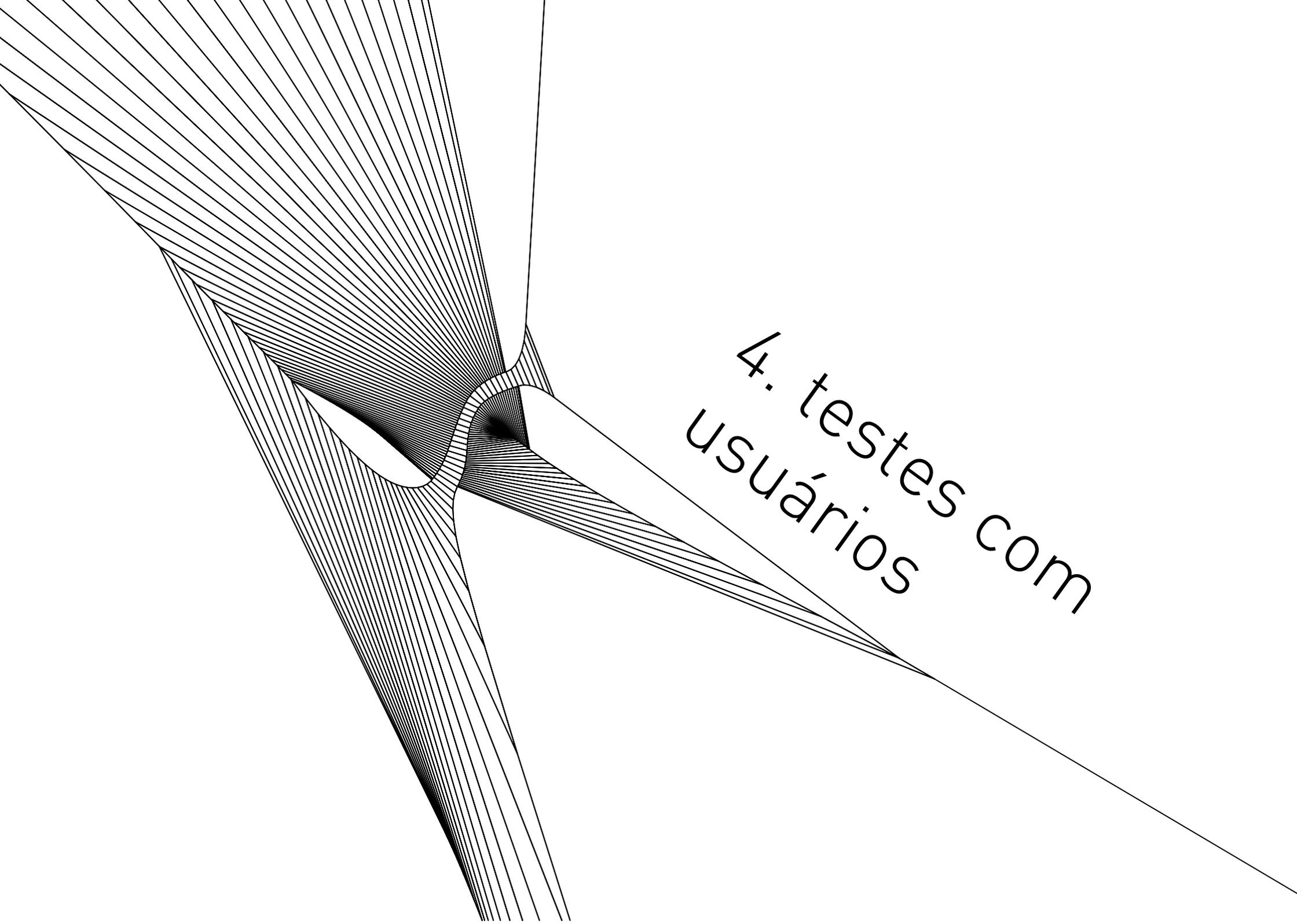
Para fazer essa escolha foram analisados os experimentos executados. Os dois primeiros “Controlador” e “Fita” foram excluídos, como foram usados de partida para os demais, não tinham bom nível de interação e de possibilidades para servirem de base. Já os experimentos com primitivas apresentavam um bom caminho, mas não foram selecionados por não gerarem objetos virtuais unidos em uma única forma, uma vez que, eram várias primitivas sobrepostas umas sobre as outras, sendo uma distribuição de sólidos no espaço não modelagem da forma que buscamos.



45. Relação do usuário com o modelo

Existiu neste momento uma divisão sobre qual caminho seria escolhido; ou o experimento das faces poligonais, que representa uma inovação mais nítida, por abordar a modelagem em sua excelência ou seguir um caminho mais próximo da simplicidade com o “escultor”, porem já abortado por outros similares como “Media Molecule’s PS4 Tech Demo” e “sketch furniture”.

Procuramos o professor Jorge lopes pedindo orientação, devido a sua vasta experiência com 3D. Ele apoiou a proposta do Quinto experimento, “Escultor”, que acabou sendo o escolhido ,uma vez que, é o mais simples para uma pessoa com pouca experiencia em 3D manusear e chegar em resultados relevantes, este também aborda uma série de questões sobre percepção de um espaço virtual, que possibilista a implementação de vários recursos de interação diferentes dos correlatos, em grande parte devido ao Kinect.

An abstract geometric drawing composed of numerous thin, overlapping lines that create a sense of depth and movement. The lines radiate from a central point, forming a complex, layered structure. A small, dark, circular spot is located at the center of this structure. The overall effect is that of a dynamic, three-dimensional form.

4. testes com  
usuários

## 4 Testes com usuários

### 4.1 Primeiro teste

Na primeira fase de testes utilizamos o quinto experimento sem nenhuma modificação e coletamos alguns dados para guiarem as alterações da aplicação.

#### 4.1.1 Feedback

A instalação não tinha nenhum tipo de feedback para o usuário se orientar sobre o que estava ocorrendo e quando não existia nenhum usuário calibrado a tela mostrava figuras desconexas.

Inserimos uma barra de topo, onde as informações são passadas por meio de textos. Durante os momentos onde não existem usuários calibrados é exibida a logo do projeto com o fundo preto.

Tivemos dificuldades técnicas para implementar a barra, uma vez que, estávamos fundindo um ambiente 3D com um 2D. Quando uma modelagem era executada ela ficava na frente do menu impedindo sua visualização.

A solução encontrada foi chamar uma função direta do java que permite desenhar uma camada sobre outra sem sofrer a influencia da primeira, semelhante a um “layer” do photoshop.

#### 4.1.2 Entrada de dados

Até o momento o programa estava usando um mouse para disparar a modelagem, porem decidimos estudar formas de comunicação alternativas onde o usuário não dependesse de nenhum apetrecho em seu corpo. Entretanto, o Kinect possui uma baixa resolução de captura, o que impede que ele reconheça dedos. Essa deficiência limita os comandos que podem ser feitos através das mãos.

Fizemos um brainstorming de possibilidades dentro das limitações existentes e chegamos a algumas alternativas como:

**Clicar levantando e descendo uma das pernas.**

**Clicar juntando e separando as mãos**

**Liberar a modelagem enquanto as mãos estão unidas.**

A primeira opção foi descartada por exigir que o kinect capture todo o usuário reduzindo sua área de desenho, além de ser uma movimentação não corriqueira e complexa, uma vez que o operador teria que levantar muito a perna para que esse movimento não fosse confundido com uma simples caminhada.

A segunda gerou conceito de “clap”. Onde o usuário junta e separa as mãos para liberar a modelagem e repete o movimento para interrompê-la. O método foi batizado desta forma, pois esse movimento se assemelha a uma palma. Na barra de feedback foram programadas frases de orientação: “Bata uma palma para iniciar o desenho”. “Bata uma palma novamente para interromper”.

A terceira foi batizada de “onOff” por funcionar como um interruptor, quando as mãos estão juntas o sistema está ligado e quando separadas ele desliga. Na barra de feedback foram programadas frases de orientação: “Junte as mãos para modelar”. “Separe as mãos para interromper”.

### 4.1.3 Lendo o usuário

Nos testes percebemos que as leituras do corpo do usuário estavam com muito ruído impedindo um desenho fluido. Para solucionar este problema programamos um código que faz uma média aritmética entre a leitura do frame anterior com o presente. Isso tornou o movimento mais fluido.

### 4.1.4 Calibragem da câmera

Os usuários relataram um incômodo quando executavam um desenho grande uma vez que a câmera estava muito próxima do objeto. Isso correu por que a câmera estava fixa na exata posição da cabeça. A solução foi posicionar a câmera alguns centímetros para trás e para cima do ponto original.

### 4.1.5 Altura dinâmica

Outro tópico encontrado nesta primeira leva de testes é a adequação da altura do cubo de modelagem para pessoas de tamanhos diferentes. A aplicação foi construída tomando por base uma pessoa de 1,85 m isso causou certo desconforto em pessoas de outra estatura.

A solução encontrada foi a criação de uma função que capta a altura do usuário quando ele é calibrado e posiciona o cubo em uma posição confortável.

## 4.2 Segundo teste

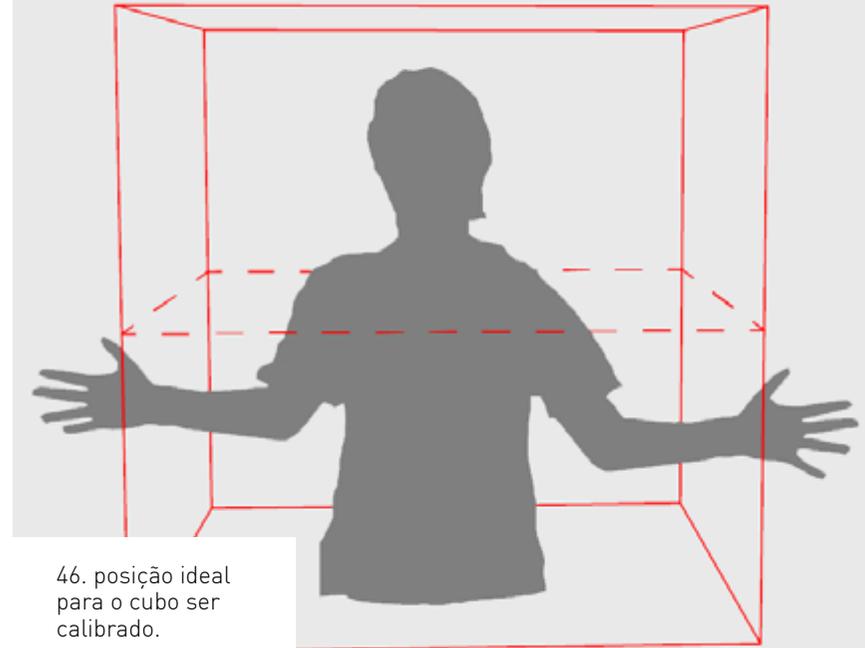
Nesta segunda bateria de testes refinamos as modificações implementadas na etapa anterior e levantamos novas questões.

### 4.2.1 Altura dinâmica

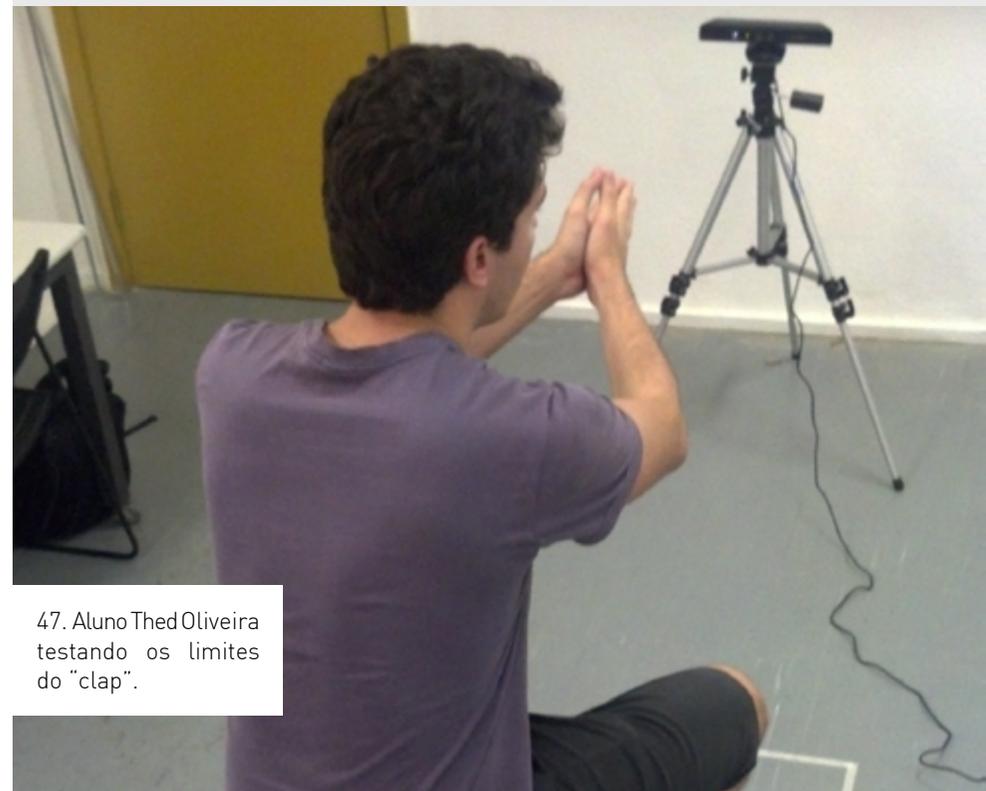
Por meio de testes encontramos a posição ideal para o cubo ser calibrado. Esta, por sua vez, deve ser a metade da altura do cubo posicionada na altura dos ombros do usuário. Como na linha tracejada da imagem.

### 4.2.2 Entrada de dados

Nesta etapa testamos as novas formas de interação “clap” e “onOff”. A primeira teve maior aceitação por meio das pessoas que já estão acostumadas com o trabalho no meio digital e depois de alguns minutos se acostumavam com o movimento, um problema observado foi, algumas pessoas executavam a palma e mantinham as mãos muito próximas o que ativava e desativava seguidamente a modelagem.



46. posição ideal para o cubo ser calibrado.



47. Aluno Thed Oliveira testando os limites do “clap”.

Uma consequência desse input é uma possível falha no desenho, no começo ou no fim da modelagem. Isso ocorre, pois o kinect perde a perfeita posição das mãos do usuário quando elas se unem, gerando assim um ruído no desenho.

O segundo método de interação, foi o que obteve a maior aceitação. O fato de unir as mãos para iniciar o desenho e separar para interromper foi de fácil assimilação por parte dos usuários, mas gerou conflitos nas leituras do kinect. Quando as mãos ficam juntas por aproximadamente 10 segundos o kinect para de reconhecer um dos braços, fato este que interrompe a modelagem. Outra limitação é que esta posição provoca a perda de orientação das mãos, gerando um ruído muito intenso na modelagem tornando impossível fazer uma linha minimamente fluida.

Após estas considerações escolhemos o método “clap”, por gerar desenhos mais fluidos.

### 4.2.3 Refinando o “clap”

Para refinar o método de entrada escolhido fomos levados a um ramo da ciência da computação conhecido como “visão computacional”. Esta ciência estuda como as máquinas capturam e processam sinais visuais e multidimensionais.

Um dos paradigmas encontrados neste campo é o reconhecimento do ponto inicial do movimento “trigger”, ou seja, em que momento o usuário deseja começar um gesto.

Neste input o trigger é o momento em que a distancia entre as mãos é zero. Mas Como foi percebido nos testes, quando as mãos ficam muito próximas um ruído é gerado no desenho. Para reduzir este efeito, criamos uma área de tolerância de 15 centímetros. Com isso o comando é realizado antes das mãos se tocarem.

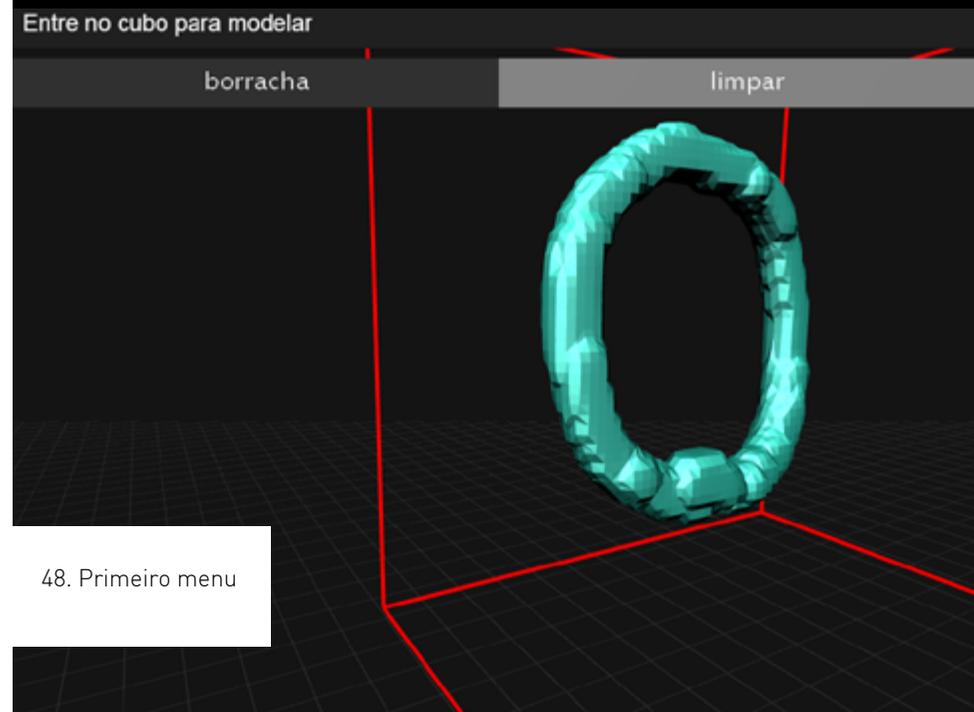
### 4.2.4 modelage apenas dentro do cubo

Abiblioteca “Toxiclibs” como foi mencionado anteriormente gera um espaço de trabalho representado pelo cubo, os modelos só podem ser executados dentro dele. Porem se a modelagem estiver ativa e a mão do modelado estiver fora do cubo o desenho é feito nas paredes deste. Isso Gera dois problemas. Primeiro, se o desenho cobrir toda a parede frontal, a visão do que está atrás é prejudicada. Segundo, quando o usuário sai da área de captura o kinect interpreta que suas mãos estão juntas, disparando assim a modelagem. A solução foi permitir a modelagem apenas quando o operador está dentro do cubo, este fica com suas arestas da cor azul. Quando o usuário sai a modelagem é interrompida e as arestas ficam vermelhas.

## 4.2.5 Primeiro menu

Até este momento o usuário não tinha acesso a nenhuma ferramenta de correção. Coletamos relatos e duas funções mostraram-se essenciais. Uma que apague tudo da tela e outra como borracha. Tivemos que tomar uma decisão. Os comandos seriam acionados por meio de gestos que representassem cada um deles ou seriam botões. Após uma pesquisa os botões mostraram-se a melhor opção. Surgiram as questões de como posicioná-los, acessá-los e ativá-los. Alguns aspectos foram levados em consideração. Devem ser nítidos, de fácil acesso e não podem atrapalhar a modelagem. Chegamos no conceito de um menu com dois grandes botões representados apenas com textos. Quando o usuário está dentro do cubo este menu some e ressurgue quando este sai.

Para ter acesso aos botões basta que o operador levante a mão direita e aponte para a opção desejada, esta por sua vez fica destacada com um fundo cinza claro. Para ativar a ferramenta selecionada basta que o usuário execute um “clap”. A escolha do “clap” como ativador foi baseada em um conceito do desenvolvimento de softwares, este recomenda que, a maneira como uma ação é executada em uma parte do programa deve ser mantida para todo ele. A opção limpar sempre aparece na direita, Já o botão esquerdo muda de comportamento. Se o pincel estiver ativo a opção borracha a parece, se esta estiver ativa a opção pincel é exibida.



48. Primeiro menu



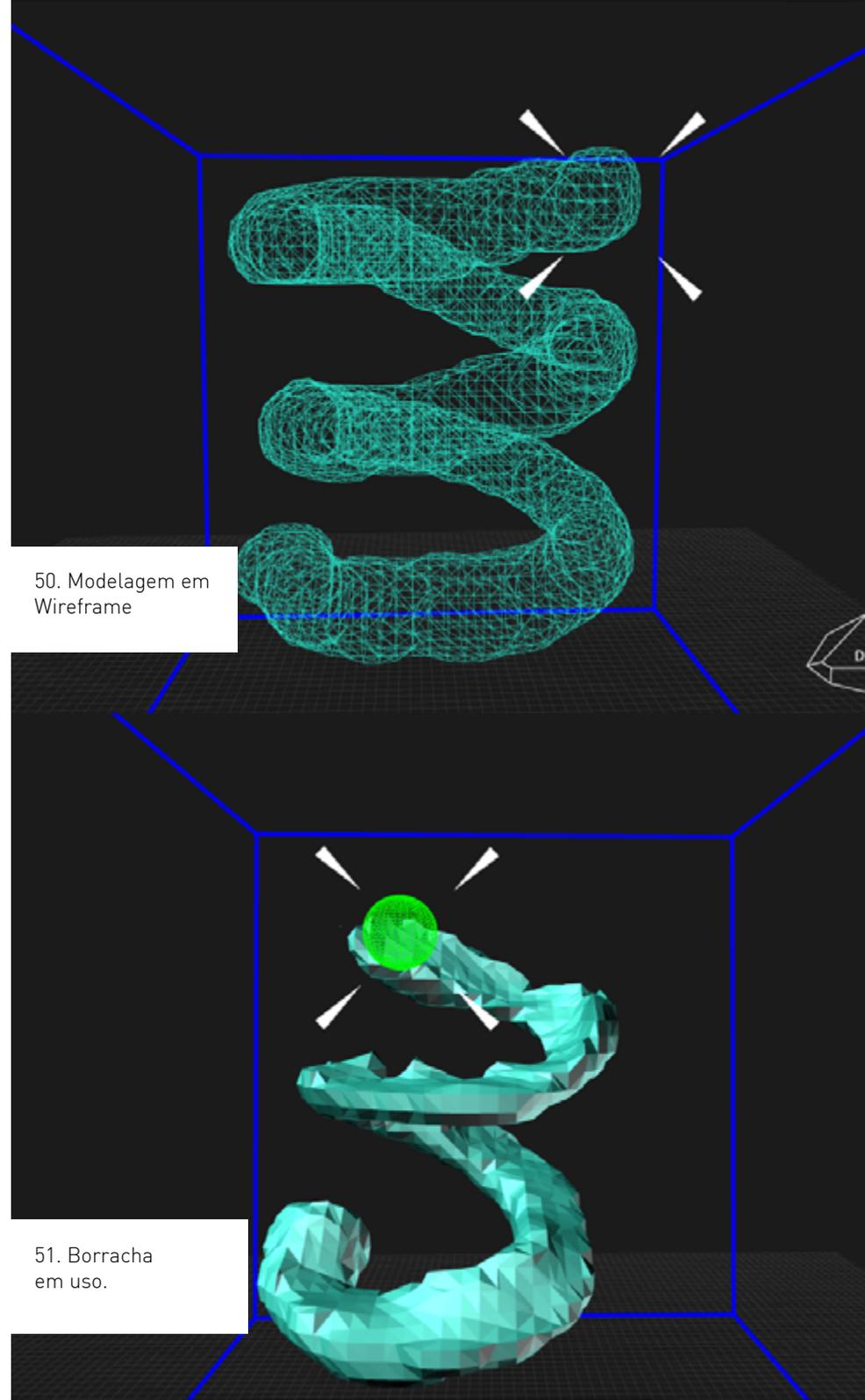
49. teste com Rafael Crespo.

## 4.2.6 Modelagem em wireframe

Nos testes percebemos alguns problemas no momento em que a modelagem está ativa. Primeiro se o usuário executa um desenho em um plano próximo de si e depois deseja desenhar algo atrás ele não consegue, pois uma parte do modelo tampa a outra. Em outra situação, usuário não sabia se a modelagem ainda estava ativa, pois os desenhos anteriores impediam a visualização do cursor. Por ultimo, a geração da malha 3D é o momento que mais exige processamento da máquina e o render com iluminação estava consumindo muitos recursos, chegando a travar a aplicação. A solução para estes problemas foi mudar o render de shade para wireframe enquanto a geração de malha está ativa.

O render pelo método wireframe representa o objeto apenas pelas linhas que compõem sem a necessidade de calcular a luz incidente. O ato de modelar tornou se mais fluido, os usuários conseguiram visualizar os detalhes que estavam atrás do modelo e ficaram cientes de quando a modelagem estava ativa ou não.

No momento em que a borracha está sendo usada a malha é vista em shade para facilitar a visualização do que está sendo apagado e o ato de subtrair partes do modelo é de fácil processamento.



50. Modelagem em Wireframe

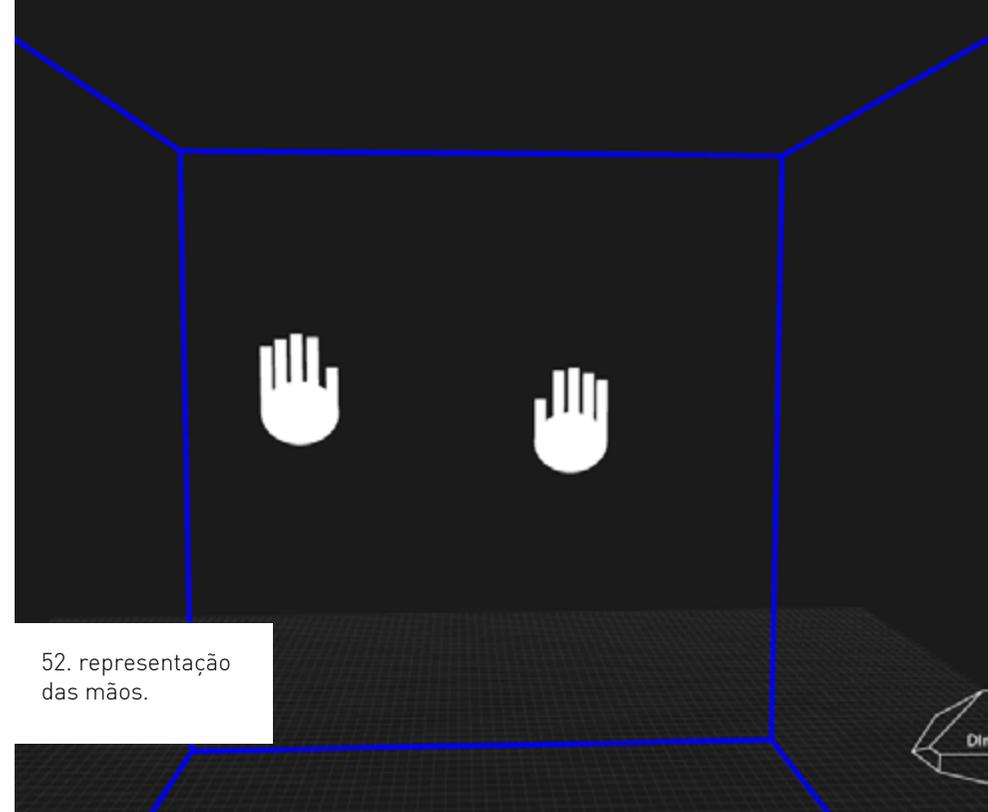
51. Borracha em uso.

## 4.2.7 Cursores

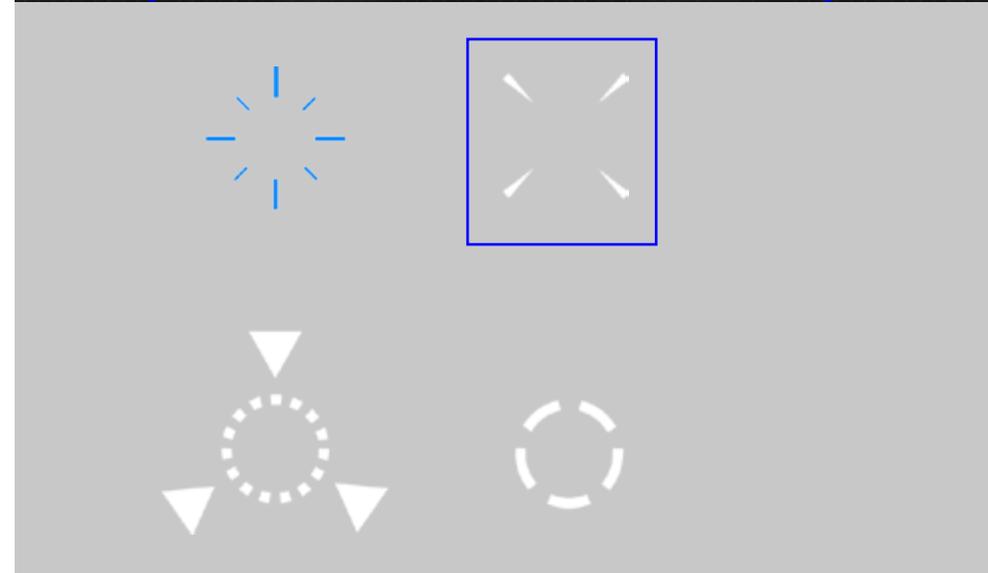
As mãos do operador estavam sendo representadas por dois quadrados, e na direita ainda existe uma esfera como “brush”. A fim de aumentar a interação e possivelmente trazer um caráter lúdico para o trabalho, os cubos foram substituídos por desenhos de mãos. Como principal inspiração utilizamos o cursor do “video game” Wii. Em nosso caso como a interface é predominantemente escura não utilizamos contorno pois o “shape” já possui bastante contraste e optamos por formas quadradas e simples para os dedos.

No momento em que a modelagem está ativa a mão direita é substituída por um desenho que se assemelha a uma mira. Como pode ser visto ao lado levantamos algumas possibilidades para este desenho e optamos pelo mais simples, baseado em apenas quatro triângulos apontando para o centro. A decisão foi feita, por ele consumir menos recursos e causar menos ruído visual.

Vale ressaltar o comportamento deste cursor em cada atividade. Quando o pincel está tivo surge a mira e de seu centro surge a malha 3D. Porém quando a borracha está ativa surge uma esfera em seu centro para facilitar o manuseio da ferramenta.



52. representação das mãos.



53. Possibilidades de mira.

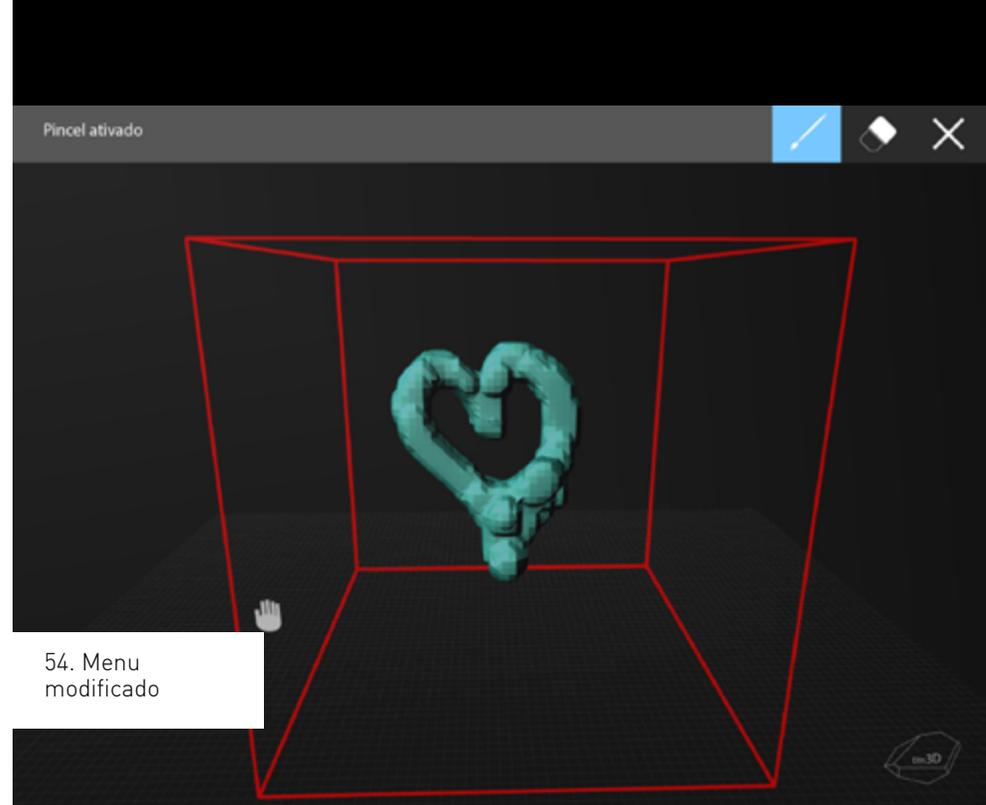
## 4.3 Terceiro teste

### 4.3.1 Revisão do menu

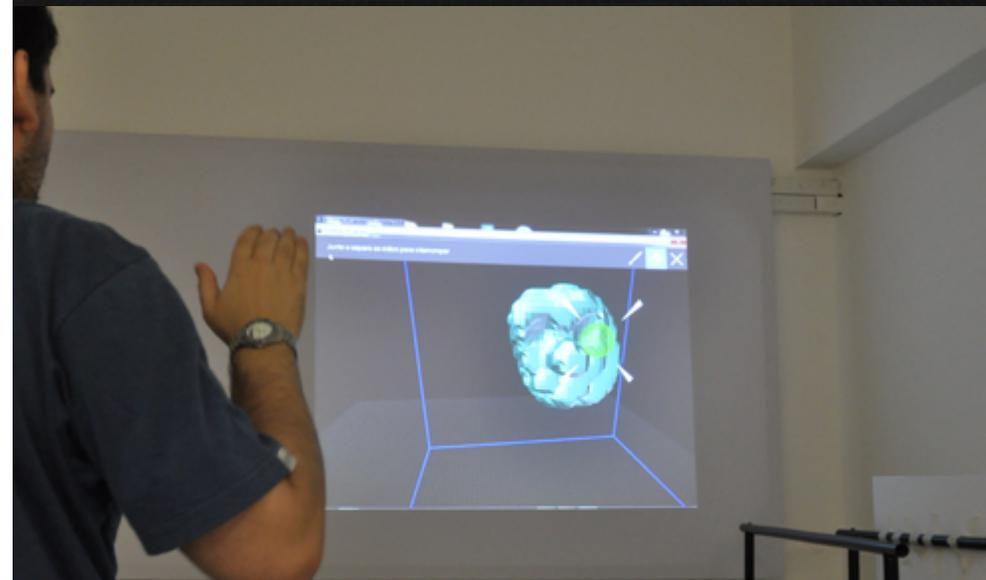
A adição do menu foi bem recebida, porém observamos algumas dificuldades em sua operação. Os usuários ficaram confusos em alguns momentos por não terem nenhum feedback visual sobre qual ferramenta estava ativa no momento e demoraram a compreender que o menu só aparece quando saem do cubo. A solução foi redesenhar a barra de topo para comportar o menu fixo. Para ocupar menos espaço substituímos os textos por ícones que se assemelham as atividades executadas. O desenho de um pincel para modelar, o de uma borracha para apagar e um X para limpar a tela. Os ícones foram dispostos no canto superior direito. A maneira de acessá-los e ativá-los continuou a mesma, o que mudou foi a visualização. Quando o menu está em uso a ferramenta selecionada fica em vermelho e depois de seu uso o instrumento escolhido fica em azul até que seja trocado.

### 4.3.2 Estratégia de modelagem

Todo programa de modelagem possui a sua linguagem e método de trabalho próprio. Com a implementação da borracha percebemos o surgimento da estratégia de modelagem do Din3D. A recomendação é que o modelador faça uma forma bruta do que deseja e depois com a borracha retire os excessos, chegando na forma que deseja.



54. Menu modificado



55. Aluno Vítor Pimentel modelando com a borracha.

### 4.3.3 Tutorial

Construímos um experimento que continha um momento tutorial, onde o software fala para o usuário o que deve ser feito. O resultado não foi satisfatório, a aplicação ficou lenta e o áudio acabou destoando do tema do trabalho. Todavia, a experimentação com áudio abriu espaço para a introdução de efeitos sonoros no programa.

### 4.3.4 Sonorização

A grande maioria dos programas inclusive sistemas operacionais, usam efeitos sonoros para dar feedback do que está acontecendo. Tomando isso como base desenvolvemos alguns efeitos com o intuito de ampliar a imersão do usuário. Fizemos uma lista com os sons que precisávamos e como eles deveriam soar:

**Click começo de modelagem:** som elevado

**Click fim de modelagem:** som descendente

**Click modelagem não permitida:** som dissonante

**Click na lixeira:** barulho metálico

**Usuário entrou no cubo:** mergulho e imersão

**Usuário saiu do cubo:** inversão do som anterior

#### 4.3.4.1 Produção dos efeitos

O som de início de modelagem, foi feito com o uso de um programa de edição de sons “MIDI”. O efeito foi gerado com o uso das notas do acorde maior de Do tocadas em sequencia. No momento da edição foram introduzidos efeitos como “phaser”, para gerar uma estética sintética, e “delay” para dar volume e profundidade ao som. O som de fim da modelagem, foi criado invertendo a ordem das notas do acorde.

O áudio de modelagem não permitida, foi obtido palhetando as três cordas mais agudas de uma guitarra, porem estas estavam abafadas. Os efeitos usados foram “reverb” e “drive”.

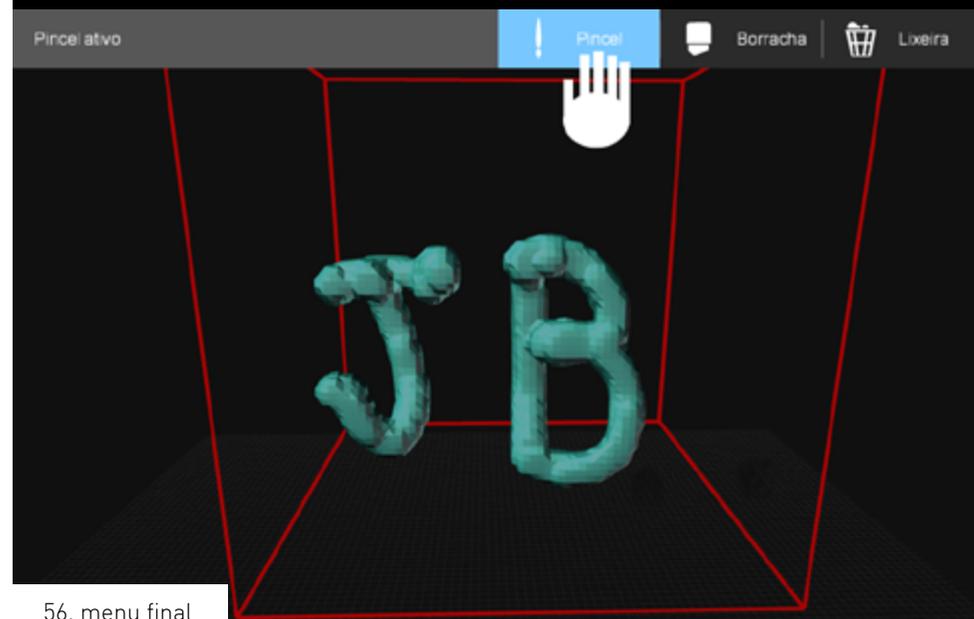
O som da lixeira foi gravado no laboratório de som e vídeo. Ele consiste do ruído gerado quando se bate uma peça metálica contra uma superfície

Os efeitos foram bem recebidos nos testes e de fato aumentaram a imersão na atividade. Outro ponto forte foi o relato de terem achado os sons divertidos e diferentes dos encontrados nos demais programas.

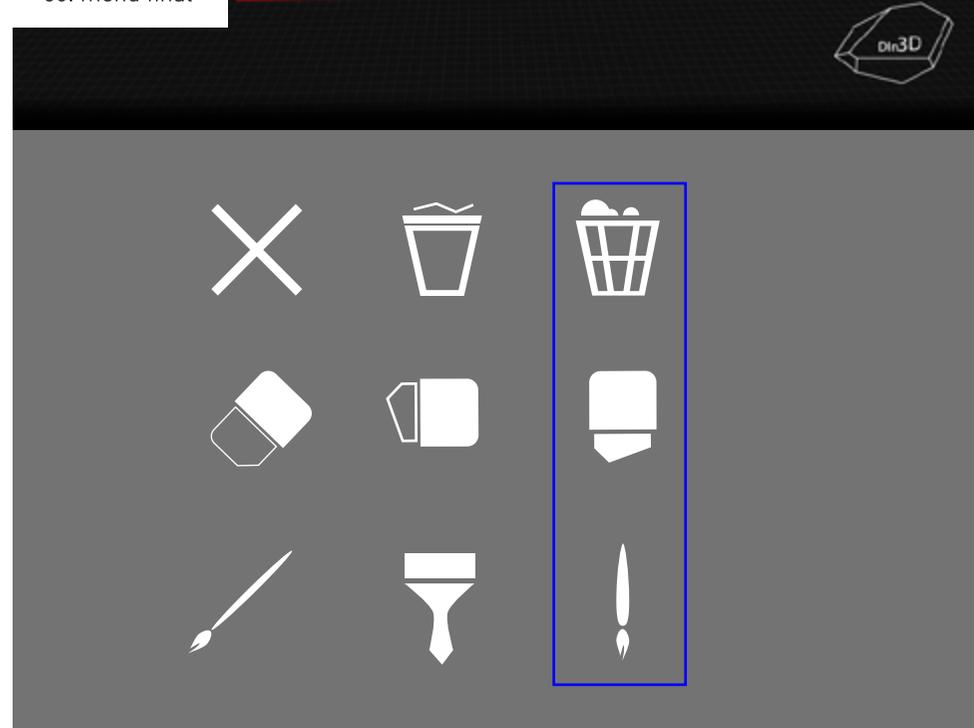
## 4.4 Quarto teste

### 4.4.1 Revisão do menu

Observando o novo menu em funcionamento vimos que estávamos no caminho correto, mas com ele novas questões surgiram. O ícone de “limpar tudo” representado por um “X” foi constantemente confundido com um comando para sair do programa. Buscando uma melhor analogia com a tarefa ele foi substituído pelo desenho de uma lixeira. Outro ponto foi o desenho dos demais ícones. Chegou a ser cogitada a modificação dos pictogramas de pincel e borracha por desenhos que não tem relação com nenhuma atividade análoga, mas sim com o conceito do que o programa está executando. Por exemplo, ao invés de usar o desenho de um pincel colocar um ícone que representasse a ideia de adição. Mas preferimos continuar com os elementos anteriores para não prejudicar a curva de aprendizado do programa. Percebemos a necessidade de alterar a rotação dos desenhos, todos estavam inclinados. Isso em resoluções baixas, como a usada neste caso 1024 x 768, causa serrilhado na imagem prejudicando sua leitura. Devido a isso todos foram redesenhados na vertical. O ícone da borracha teve o miolo preenchido, pois estava sendo confundido com um cadeado e o desenho do pincel ficou mais largo na lateral e menor na altura.



56. menu final



57. Opções de ícones

Os botões aumentaram de largura, passando a ocupar metade da barra. Ao lado dos ícones foram colocados os nomes das funções. Agora quando o menu está em uso surge o desenho da mão sobre a função selecionada, como se o usuário estivesse pegando o comando.

Uma importante alteração na forma de chamar os comandos foi realizada. Antes, para ativar uma ferramenta o usuário deveria juntar as mãos com o ícone selecionado. Isso gerou problemas na usabilidade. Por uma questão de proporção corporal, era difícil para algumas pessoas manter uma das mãos parada e juntar com a outra, em alguns casos, principalmente para ativar a lixeira, o braço esquerdo não tinha comprimento o suficiente para completar o movimento. A solução foi. O usuário simplesmente ergue o braço direito aponta para a opção desejada, quando este descer o braço a ferramenta é ativada. Esse movimento tornou a troca de funções mais simples e intuitiva.

#### 4.4.2 Curva de aprendizado

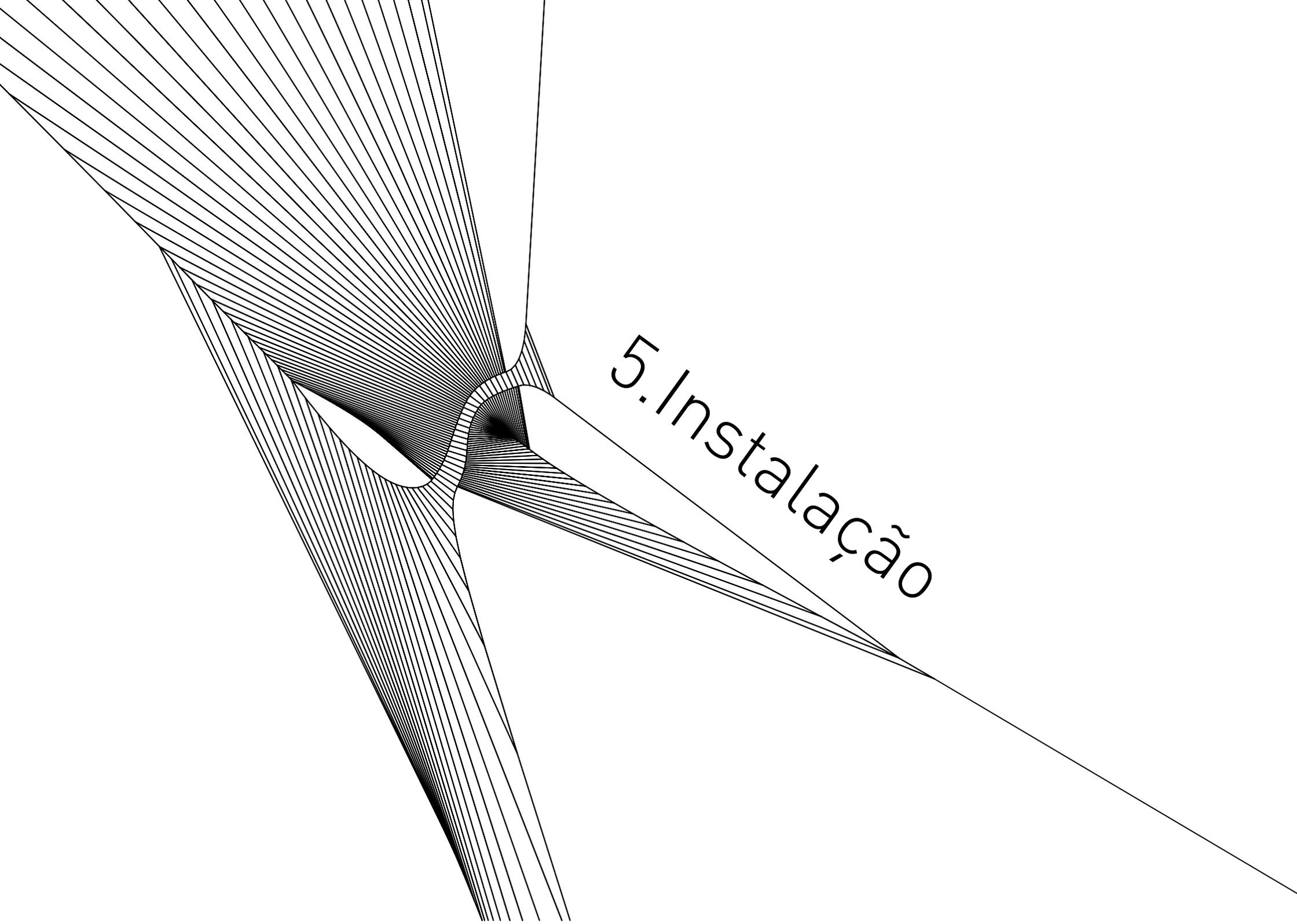
Fazendo uma consideração geral sobre todos os testes feitos chegamos a conclusão de que a curva de aprendizado do programa está bastante satisfatória, cada usuário não demorou mais do que 5 minutos para ter controle total da aplicação. E relataram que a experiência foi divertida e se sentiram confortáveis.

#### 4.4.3 Cor e iluminação da malha

Quando a modelagem não está ativa o objeto é renderizado pelo método Shade, que representa os objetos simulando a reflexão de fontes luminosas em sua superfície. Nesse método a cor dos objetos pode ser alterada sem acarretar grande perda de desempenho.

Em um primeiro momento a cor usada foi o branco, depois trocamos para marrom em uma tonalidade próxima do barro. De forma unanime pediram para que a cor não tivesse ligação com nenhum tipo de material e que também não esteja presente na interface. Foram testadas inúmeras cores até que o verde foi escolhido.

Com relação à luz. Existia apenas uma lâmpada virtual apontada para o objeto de cima para baixo e da esquerda para a direita. Isso impedia uma boa visualização quando a câmera estava a direita da cena. Isso foi resolvido inserindo uma segunda lâmpada na posição oposta a primeira.

An abstract geometric design composed of numerous thin, overlapping lines that create a sense of depth and movement. The lines radiate from a central point, forming a complex, layered structure. A prominent circular shape is formed by the intersection of these lines, creating a dark, shaded area. The overall composition is dynamic and modern.

## 5. Instalação

## 5. Instalação

O projeto em sua concepção tinha o intuito de desenvolver uma instalação interativa com os resultados dos experimentos. Porém, pela maneira com que os testes de usabilidade influenciaram a aplicação e pelos relatos coletados, percebemos o potencial de transformar a instalação em um software.

“DIn3D é um software de modelagem tridimensional para profissionais, estudantes ou amadores que buscam uma nova maneira de interagir com o ambiente de trabalho virtual.”

A instalação ganhou um caráter de promoção e divulgação do software em feiras e mostras de tecnologia.

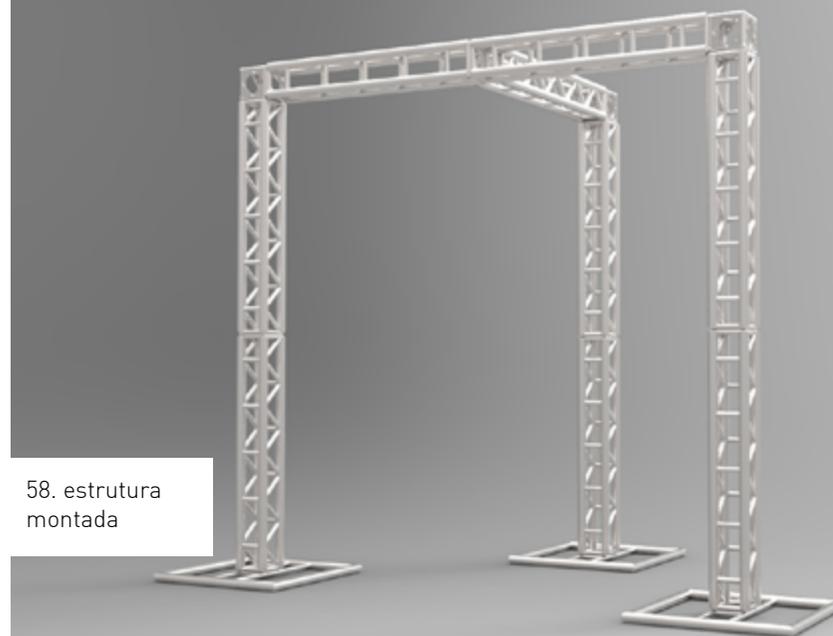
### 5.1 Desenho da instalação

Devido a um bug já solucionado do programa, o kinect não poderia localizar nenhum outro usuário, caso contrário ele travaria. Isso afetou diretamente o desenho da instalação. Esta deveria ser uma sala que possibilitasse a visualização da tela pro outras pessoas mas que limitasse o número de usuários.

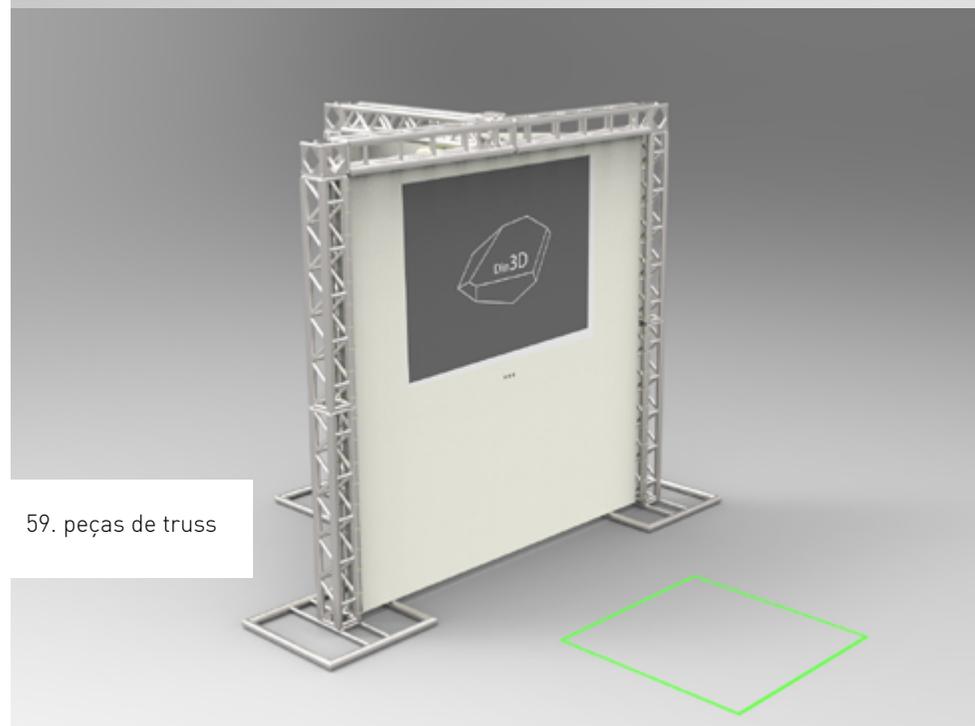
A estrutura planejada foi uma sala em forma de U fechada. Onde as paredes seriam feitas com uma estrutura de madeira revestida por lona. Uma vez com o bug solucionado e a aplicação comportando-se bem com a presença de vários usuários, simplificamos o desenho da instalação para um pequeno corredor formado por duas paredes, com o intuito de facilitar a visão de quem está de fora. As estruturas seriam feitas com esqueletos de cano ou de esquadrias de alumínio desmontáveis, também revestidas por lona. Em ambos os desenhos o planejamento era sustentar o projetor por vigas ou prende-lo no teto, mas isso além de encarecer o projeto não é uma solução elegante.

Como está é uma instalação para exposições encontramos uma estrutura comum nesse tipo de evento o “Boxtruss”. Esse é um sistema modular que permite a montagem de estruturas para diversas finalidades, como suporte para iluminação e palanques de shows. O desenho final foi feito tomando como base o minitruss Q15. A estrutura consiste em três pilares formando um totem triangular revestido pro lona. A projeção é feita por “backprojection” usando um projetor de curta distancia que possibilita projeções de cima para baixo, com uma distancia de 40cm da tela. O kinect fica preso e escondido dentro desta estrutura por uma barra fixada entre duas pilastras, para que ele veja o ambiente serão feitos três furos circulares na altura dos sensores. Os demais componentes como computador, fonte e caixas de som ficam posicionados dentro do totem.

Um metro a frente a projeção será desenhado um quadrado que representa a área de modelagem. Este será feito com fita vinil fotoluminescente, que é usada em salas de cinema. Cogitamos a possibilidade de usar uma fita de LEDs para esta representação, mas isso exigiria o uso de uma canaleta, que poderia causar acidentes.



58. estrutura montada



59. peças de truss

## 5.2 Montagem

### Itens necessários:

9 colunas truss de 1 m

3 corners

4 braçadeiras

3 bases

3 ganchos

1 barra

1 pacote com 500 braçadeiras de nylon

### Equipamentos:

1 Kinect

1 base de kinect para tripé

1 computador ( com os periféricos de entrada)

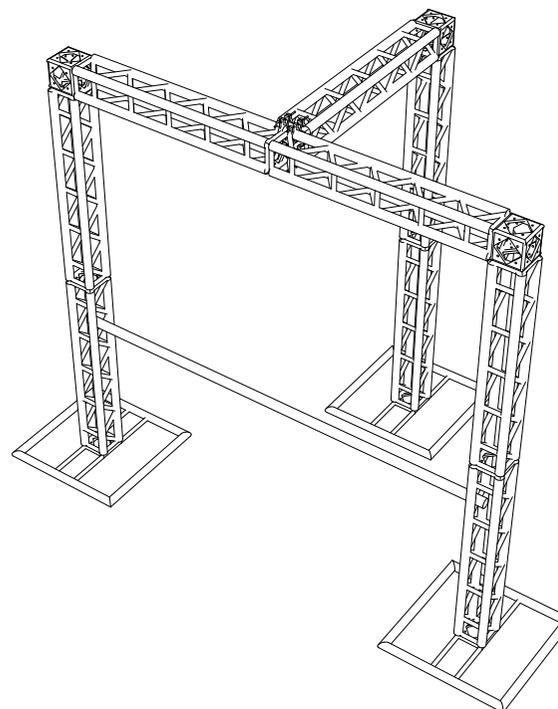
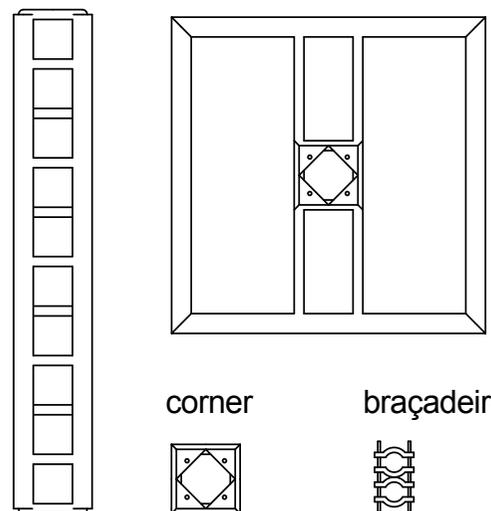
1 estabilizador

2 ou 4 caixas de som satélite

1 projetor de curta distancia

A estrutura é formada por uma trave de 2 x 2 m e uma meia trave de 1 x 2 m. Esta é fixada na trave por meio das braçadeiras. O kinect é preso na barra.

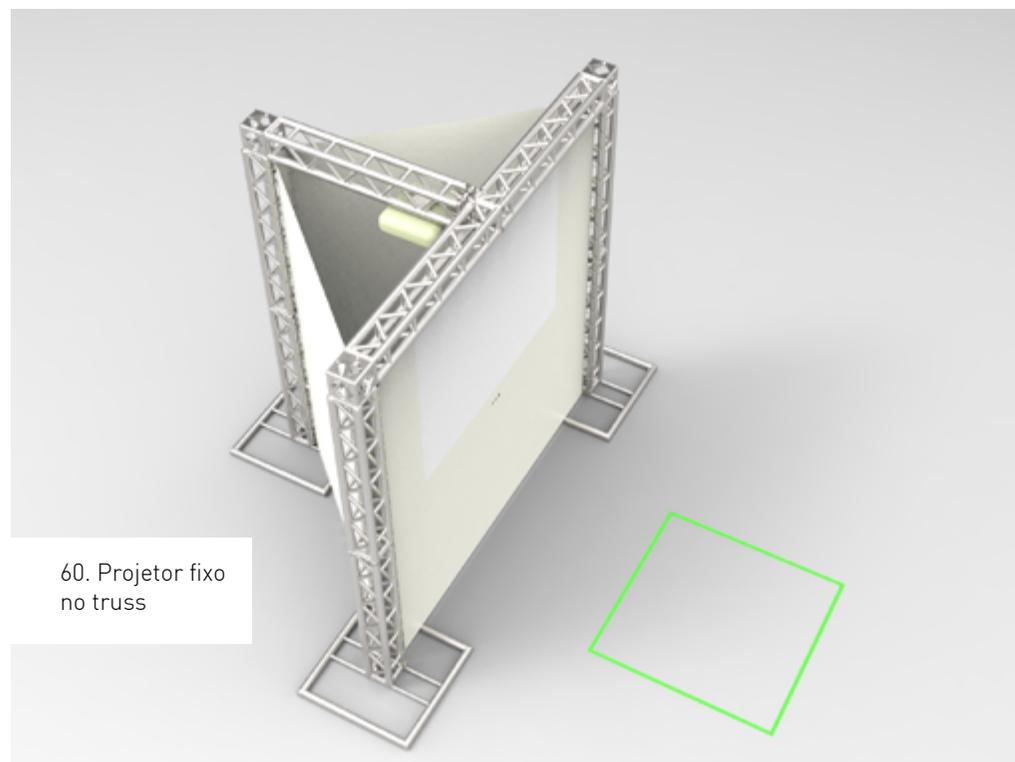
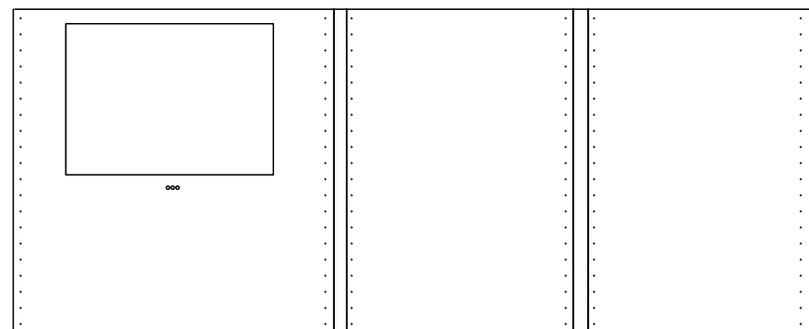
coluna Base

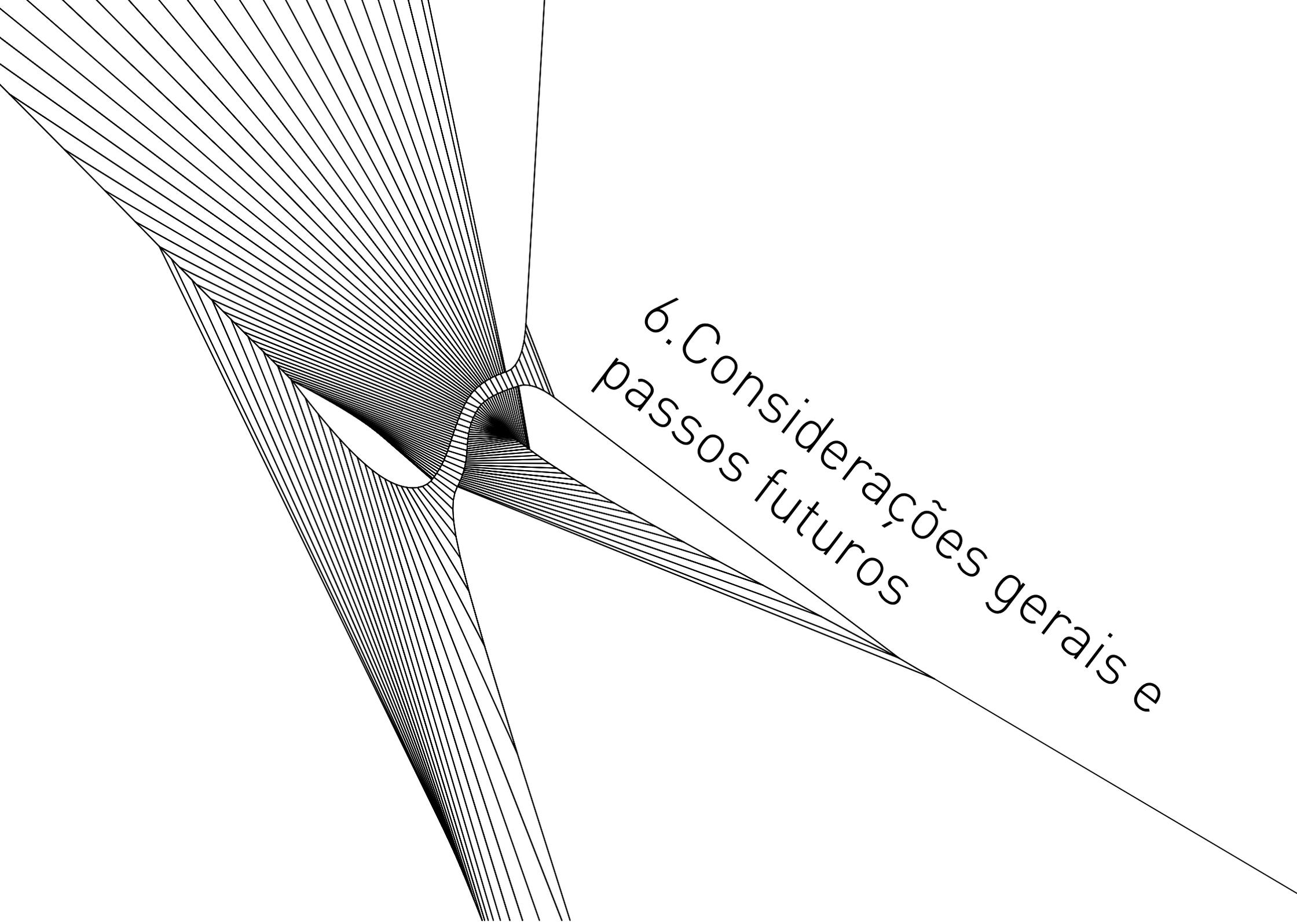


A lona que reveste a estrutura possui cinco metros de comprimento. Ela é fixada na coluna por meio de braçadeiras de nylon, que passam por furos no tecido. Para aumentar a durabilidade da malha os buracos são protegidos com ilhoses.

As caixas de som ficam presas nas duas colunas laterais. No caso de apenas duas elas devem ficar a 1,5m do chão. No caso de 4 as duas outras ficam a 0,5m.

Previsão de orçamento diário R\$ 288,60 estrutura de truss, R\$ 300,00 projetor e R\$150,00 demais equipamentos. A lona pode ser reutilizada e sai por R\$120,00.



An abstract geometric drawing composed of numerous thin, black lines. The lines are arranged in a complex, overlapping pattern that forms a central, dark, circular void. From this central point, several distinct, fan-like structures radiate outwards, each formed by a series of parallel lines that converge towards the center. The overall effect is one of depth and intricate detail, resembling a stylized architectural or organic form. The text is positioned in the lower right quadrant of the image, following the curve of one of the radiating structures.

6. Considerações gerais e  
passos futuros

## 6.Considerações gerais e passos futuros

A etapa final do projeto representou a busca pelo equilíbrio entre as alternativas de criação e a realidade de produção. As limitações técnicas tiveram profundas influencias nas decisões tomadas e é neste momento que entra o “design de interação”, através dele buscamos a melhor forma de comunicar homem e maquina levando em conta cada um dos lados.

Fico muito feliz de ter chegado no fim de um ano de desenvolvimento com a versão beta de uma programa que gerou comentários como “gostaria de ficar brincando o dia todo com isso!” e “- Sempre quis modelar direto com as mãos!”. Tive muitas dúvidas no meio do processo se daria certo, mas com boas orientações e vencendo uma dificuldade por vez o caminho foi trilhado. Este projeto ainda tem muito que evoluir e não se encerra com o fim desta matéria. Pretendo manter o desenvolvimento deste software implementando novas funções e sempre focado na inovação e simplicidade. Até a data da entrega deste relatório ainda não avia sido lançado o novo modelo de Kinect. Com este equipamento, que possui maior resolução, poderei implementar o click diretamente pelo ato de abrir e fechar a mão e projetar interações por meio dos dedos.

O software será reescrito em uma linguagem que possibilite melhor desempenho e a geração de malhas mais complexas. Serão implementadas ferramentas que usam por base a generatividade e a simulação de física para gerar modelos, conceitos estes ainda pouco explorados pela indústria.

# 7. Referências Bibliográficas

## Teses

MAURO PINHEIRO, R. Design de interação e computação pervasiva. Rio de Janeiro, 2011. 76-77p. tese (doutorado) - Faculdade de design, Pontifícia universidade católica do Rio de Janeiro.

## Livros

WANDS, B. Art of the digital age. 1.ed. Thames & Hudson, 2006.

FREYER, C.; NOEL, SS.; BUCKI, E.; Digital by Design: Crafting Technology for products and environments. 1.ed. Thames & Hudson, 2010.

SCHWARTZMAN, MADELINE. See yourself sensing: redefining human perception. 1.ed. Black Dog publishing, 2011.

HALL, E. talk to me: design and the communion between people and objects. 1.ed. New York: Moma the Museum of Modern Art, 2011.

GALANTER, P. What is Generative Art? Complexity Theory as a Context for Art Theory Disponível em:<http://philipgalanter.com/research/>. Data de acesso: 08/09/2012

## Revistas digitais

Hewett; Baecker; Card; Carey; Gasen; Mantei; Perlman; Strong and Verplank . Human-Computer Interaction Disponível em:[http://old.sigchi.org/cdg/cdg2.html#2\\_1](http://old.sigchi.org/cdg/cdg2.html#2_1). Data de acesso:08/10/2012

Reas. C. Disponível em:<http://reas.com/>. Data de acesso:21/09/2012

\_\_\_\_\_. high-tech low-tech Disponível em:<http://hlt.media.mit.edu/>. Data de acesso:21/09/2012

Reas. C. How to draw with code Disponível em:<http://vimeo.com/44410722>. Data de acesso:30/08/2012

Reas. C. How to draw with code Disponível em:<http://vimeo.com/22955812>. Data de acesso:13/09/2012

NORMAL. N. Build a Touchless 3D Tracking Interface with Everyday Materials Disponível em:<http://blog.makezine.com/2012/08/10/build-a-touchless-3d-tracking-interface-with-everyday-materials/>Data de acesso: 10/09/2012

Z33. Artisan Eletronique Disponível em:<http://unfold.be/pages/5/items/90>. Data de acesso: 27/08/2012

BHATNAGAR. R. Modeling sound in 3D with voice extruder Disponível em:<http://blog.ponoko.com/2011/10/12/modeling-sound-in-3d-with-voice-extruder/> Data de acesso: 23/08/2012

BHATNAGAR. R. Modeling sound in 3D with voice extruder Disponível em:<http://blog.ponoko.com/2011/10/12/modeling-sound-in-3d-with-voice-extruder/> Data de acesso: 23/08/2012

RIBEIRO. T. Sequência de Fibonacci Disponível em:<http://www.infoescola.com/matematica/sequencia-de-fibonacci/> Data de acesso: 26/11/2012

BRUSAMARELLO. V. Sensores de temperatura Disponível em:[http://www.chasqueweb.ufrgs.br/~valner.brusamarello/inst/aula\\_06.pdf](http://www.chasqueweb.ufrgs.br/~valner.brusamarello/inst/aula_06.pdf) Data de acesso: 26/11/2012

CIRCUIT. O que é um protoboard? Disponível em:<http://estudeltronika.blogspot.com.br/2009/07/o-que-e-um-protoboard.html> Data de acesso: 26/11/2012

.