

# Interactive Visualization of the CMOA

1<sup>st</sup> Kayo Yokoyama  
FGV EMAp  
Rio de Janeiro, Brazil  
kayo\_reis1@hotmail.com

1<sup>st</sup> Raul Lomonte  
FGV EMAp  
Rio de Janeiro, Brazil  
raullomonte13@gmail.com

1<sup>st</sup> Vinícius Hedler  
FGV EMAp  
Rio de Janeiro, Brazil  
viniciushedler@gmail.com

**Abstract**—Este artigo apresenta um projeto de visualização de dados das artes do Carnegie Museum of Arts [1], explorando a riqueza da coleção e fornecendo insights interessantes sobre as obras e os artistas representados. Com isso, o projeto implementa recursos interativos para os usuários explorarem a coleção, permitindo que eles descubram conexões e padrões entre as obras e mergulhem na diversidade cultural retratada.

**Index Terms**—data visualization, interactivity, arts

## I. INTRODUÇÃO

No mundo moderno cada vez mais digitalizado, onde a arte pode ser encontrada em diversas formas e em inúmeras mídias digitais, a visualização de dados se apresenta como uma ferramenta poderosa para nos reconectar com o aspecto de explorar a arte. Com isso, através de visualizações, é possível criar experiências interativas e imersivas que permitem aos usuários mergulhar em um universo de expressão artística. Assim, a abordagem apresentada busca proporcionar uma nova maneira de apreciar as artes, ampliando os horizontes e incentivando a exploração de diferentes estilos, técnicas e artistas. Além de tudo, queríamos expandir a criatividade e sair do mundo das visualizações que se ficam limitadas por eixos e números.

## II. NO MUNDO DOS DADOS

### A. Base de dados

Então, após certa procura para uma base de dados que pudesse se encaixar no projeto que estávamos pensando achamos parte da coleção do CMOA (Carnegie Museum of Art's) [2], um Github e uma postagem no Kaggle, que apontava ser basicamente a mesma coisa encontrada no github apesar da publicação ser mais recente. Tendo feito essas escolhas, partimos assim para a limpeza e o refinamento dos dados encontrados nos datasets e como era de se esperar de uma base disponibilizada à 8 anos, algumas informações estavam incoerentes. Sendo assim, tivemos que consertar, ou desconsiderar algumas das informações.

### B. Tratando os dados

Primeiramente, percebemos que:

- Os links que apontavam para a página ou imagem das artes estavam desatualizados
- Algumas das células no dataset principal continham valores nulos (null, NaN, none)
- Havia itens duplicados na base.. ou quase

Dados esses pontos, começamos o tratamento em *Python* e para isso foram usados as bibliotecas *Pandas* e *Selenium*. *Pandas*, para ter um fácil manuseio do csv contendo as informações e *Selenium* pois ele consegue extrair informações de sites que renderizam a página de forma dinâmica com o uso de *Javascript*, como é feito pelo CMOA. Partindo disso, fizemos um processo de extração dos links das fotos que precisaríamos para uso nas visualizações dado que o resto já estaria no arquivo csv, além disso também explicaremos futuramente que não usamos o tamanho original das imagens e sim um reduzido. Pelo fato de precisar desse acesso à página com *Selenium* para conseguir as imagens, requisitadas de oitenta mil links, a extração levou alguns dias. Só foi possível acessar todas as imagens, pois em nossa base de dados tinha uma coluna que representava o id da arte, que juntando a uma parte fixa de link chegava a exposição no site, onde conseguimos pegar a imagem. E, no geral foi substituído o link original das imagens para um mais atualizado e com figuras de tamanho menor.

1) *Para o CMOA Fragments*: Foram retiradas ou reformataadas as células que não tinham valores, como títulos que estavam como "null" mas que deveriam ser "Untitled", ou linhas com itens duplicados, mas haviam vários casos em que as obras eram distintas contudo tinham o mesmo design por terem sido feitas pelo mesmo artista. Um exemplo de item que a primeira vista parece ser duplicado é o "Jug" de "Dresser, Christopher/Old Hall Earthenware Company Limited", mostrado na Figure 1.

2) *Para o CMOA Timeline*: No caso do CMOA Timeline foi necessário acessar a data das imagens, porém a coluna de data possuía valores com formatações diversas. Então utilizando-se regex através de python foi criada uma nova coluna com apenas um número inteiro representando o ano da obra. Este número precisou ser estimado nos casos em que havia um range de data (ex: "1900-1915") ou uma parte do século (ex: "late 19th century"). Nos casos onde não foi possível coletar a data ou a mesma não estava presente o valor da coluna foi marcado como "999999", colocando-os ao final da visualização. Outra demanda que surgiu com esta visualização foi a de filtrar os artistas por quantidade de obras registradas, mantendo apenas os top 200. Isto porque muitos artistas possuíam poucas obras e obras sem data, tirando a efetividade da visualização, além de que tais artistas possuíam uma chance maior de não ter um artigo bibliográfico



Fig. 1: Jarros com mesmo design, porém diferentes

acessível. Em algumas versões também foram removidos as obras por artistas desconhecidos (ex: "unknown american"), porém posteriormente foi percebido que tais obras mostram uma história interessante e relevante sobre a arte no país ao longo dos séculos.

3) *Para o CMOA Treemap*: A fim de se chegar na visualização final, foi preciso algumas manipulações diretas no dado. Não era viável mostrar todas as imagens de todas as categorias, ficava extremamente poluído e muito lento para carregar todas as imagens, portanto, a opção que encontramos foi a de diminuir o número de imagens expostas, porém mantendo a área do treemap referente a quantidade total de imagens daquela categoria e país, assim continuamos passando a informação que queríamos codificando com a área e mostramos uma quantidade razoável de imagens. Caso tenha mais que 20 imagens em determinada nacionalidade e classificação, são selecionadas apenas 20, essas 20 foram escolhidas de forma aleatória.

### III. ESCOLHAS DE DESIGN

#### A. CMOA Fragments

1) *Motivação*: A principal motivação para esta parte do trabalho foi conseguir mostrar o quão vasta e variada é a quantidade de artes que se tem pelo globo. Com isso, uma ideia inicial vista para ser feita foi um grid infinito de imagens que pudesse ser arrastado tanto no eixo x como no eixo y como feito aqui Infinite Grid, porém essa ideia se tornou muito complicada e pouco intuitiva para o propósito da visualização. Além disso, outra inspiração foi que convencionalmente em museus as pessoas se limitam a visualização de uma arte por

vez no geral, mas aqui tendo postas elas em um grid há um maior aumento sensorial pela quantidade de artes, não tantas que não de para se apreciar e nem tão poucas que não de para complementar a ideia de grandiosidade.

2) *Implementação*: Primeiramente, vamos falar que ferramentas foram usadas: HTML, CSS, Javascript, juntamente com D3.js, jquery.js, isotope.js e Python para formular dados, dito isso, partimos para a implementação. Assim, foi feito uso do isotope para cuidar da disposição das imagens no layout mostrado no site, um grid de imagens que pode ser rearranjado com 2 botões, e para complementar a ideia de grandiosidade dita anteriormente um scroll infinito foi implementado com js. Desse modo, tendo essa base pronta a próxima tarefa foi fazer com que o usuário pudesse apreciar a vastidão das imagens, mas que ao mesmo pude observar uma arte específica com mais detalhes, para tal, um zoom e um hover foram feitos com js para mostrar a imagens com mais clareza e para exibir informações da obra como título, artista, data e classificação/estilo de arte. Além disso, queríamos dar mais liberdade para que o usuário pudesse explorar de mais formas a coleção, com esse propósito em mente um botão para filtrar pela classificação da arte foi implementado, juntamente com um botão para mostrar as imagens de forma aleatória e outros dois para mudar o layout exibido, todas feitas em js.

Ao longo do processo alguns eventos que mais geraram obstáculos para chegar na visualização final foram a questão de saber e executar a forma mais apropriada de exibição das imagens, o tratamento e extração dos dados necessários, que ferramentas utilizar e alguns bugs que apareciam ao longo do caminho. O primeiro empecilho foi por conta de termos tentado confeccionar um grid infinito, mas sem sucesso depois de muitas tentativas e principalmente por que o código base usava imagens de tamanho igual, o que não era possível para o nosso caso. Seguidamente, limpar e refatorar os dados como foi dito anteriormente. Para mais, saber e encontrar as ferramentas adequadas foi um desafio no geral pois para achar a ideal precisasse de tempo para pesquisar e testar a todas as opções descobertas para por fim selecionar a mais acertiva. Além de tudo, bugs diversos que ocorriam por conta de conflitos ou falta de experiência.

3) *Absorção de ideias*:: Ademais, críticas, ideias e comentários foram feitos para melhorar a visualização como um todo. Estes foram:

- "Na página de fragmentos eu sugiro que adicione um pequeno texto ao lado da imagem quando clicada e o hover mostre o nome do autor/ano.": Foi implementado para a visualização final
- "...caso pareça razoável, sugiro que o usuário possa buscar a arte que deseja visualizar. Digo isso porque me parece que a distribuição das imagens é aleatória a cada clique (a cada vez que você abre a página, as imagens que surgem inicialmente são diferentes)": Uma lista para escolher por classificações foi feita para resolver junto ao scroll infinito para exibição de mais imagens.

- "... poderia haver um texto explicando o que está sendo mostrado na página.": Um hover para exibir informações da obra foi implementado.
- "...O usuário poderá ficar perdido, pois não sabe se as obras estão agrupadas por similaridade ou não..." e "... Um scroll infinito, como comentado em aula, poderia agregar mais.": Um dropdown para escolher a classificação foi feito e o scroll implementado.
- "Existe um bug na página de CMOA Fragment onde quando você fecha uma imagem e rapidamente abre outra a imagem fechada aparece com um alpha baixo em cima da imagem que está sendo mostrada...": A mudança de opacidade das imagens foi retirada já que com o zoom já é possível ter um foco maior em alguma obra específica.

## B. CMOA Timeline

1) *Motivação*: A ideia para esta visualização começou com a inspiração na visualização Substate [3], que mostra justamente a evolução de algo no tempo, juntando com o scroll de uma parte da tela enquanto um conteúdo complementar, explicativo acompanha o usuário em outra parte da tela. A motivação da visualização foi então o usuário conseguir ver as obras ao longo do tempo, podendo comparar os estilos e características delas. A ideia inicial então foi fazer uma linha do tempo geral, com todas as obras contidas no acervo.

Ao longo do projeto a ideia foi mudando, passando por mostrar apenas algumas obras por período, mostrar apenas o artista mais ativo em cada período e uma de suas obras, e ao final chegando em permitir que o usuário escolha um artista em específico e veja somente as suas obras enquanto também tem acesso à um pedaço de sua biografia.

Espera-se que o usuário possa explorar obras e categorias nas outras visualizações do site e uma vez que tenha encontrado um artista que lhe interesse, possa vir para esta visualização e conhecer as obras do artista, bem como entender um pouco sobre sua história e se aprofundar em sua arte se conectando com seu traço e filosofia ao longo do tempo.

2) *Implementação*: Como mencionado na seção de tratamento de dados, a primeira etapa (não cronologicamente) para a construção da visualização consistiu de fazer o tratamento dos dados filtrando os artistas e criando a coluna de data numérica. Então partiu-se para a criação da visualização estática e depois para a adaptação da mesma. Foi utilizado d3 para que as imagens fossem inseridas de acordo com a base de dados e para que o seletor contivesse o nome de todos os artistas de maneira dinâmica.

Houve um local onde a parte dinâmica não foi feita através de d3, que foi o texto da biografia. Este texto é coletado utilizando-se a api da Wikipedia [5]. Um problema encontrado neste momento foi o de que algumas vezes um autor possui um nome levemente diferente na base de dados e na Wikipedia, seja por erro em algum dos lados (ex: "Harry" na base versus "Henry" na Wiki) ou por obstrução de sobrenomes ou nomes do meio (ex: "George W."). Isto impossibilitava de fazer um acesso direto por url, então foi necessário utilizar a api para fazer uma busca, acessando então a página retornada

na primeira posição (melhor resultado de acordo com a api). Ainda assim eventualmente isto não era o suficiente, levando a páginas erradas e, consequentemente, biografias erradas. Então decidiu-se utilizar de técnicas de programação de linguagem natural, mais especificamente o cálculo de distância entre duas sentenças, para calcular a similaridade entre o artista procurado e a página retornada.

Foi calculada a Distância de Levenshtein entre o nome do artista alvo e o título da página encontrada. Caso a distância seja menor do que um certo grau, a página é utilizada e a biografia inserida. Caso contrário, no lugar da biografia é inserido um texto relatando que não foi possível encontrar uma página sobre o artista.

3) *Absorção de ideias*: As sugestões referentes à visualização da linha do tempo estavam bastante relacionadas à parte estética e de disposição das imagens. E realmente, na versão apresentada anteriormente, as imagens estavam dispostas em um scroll horizontal e em um bloco pequeno, além de terem um frame que comprometia sua natureza. Como sugerido, esta disposição foi alterada, dando muito mais espaço para a visualização, removendo o frame arredondado e incluindo um frame que valoriza mais a obra sem alterá-la, e mudado o scroll de lateral para vertical, de maneira que seja aproveitado o scroll natural da página, deixando a experiência do usuário mais fluida.

Uma das sugestões se preocupava com o caso de um artista possuir várias obras em um determinado ano. Não foi feito nenhum tratamento especial para este caso, mas sim foram mostradas as diversas obras deste ano dispostas tal qual deveriam em relação as obras de outros anos. Então acabou que o problema do qual a sugestão tratava não se concretizou, e a sugestão até foi implementada mas simplesmente através do curso natural da visualização.

## C. CMOA Surface

1) *Motivação*: Um objetivo do trabalho era mostrar a quantidade de imagens relacionando a nacionalidade, assim passando uma ideia geral da cultura do país, visualizando suas artes de forma geral. A área é um ótimo encode de quantidade, portanto a ideia inicial foi a de fazer um gráfico de barras, mas pensamos que como estamos trabalhando com obras de artes, o treemap [4] seria perfeito, pois conseguiria além de mostrar a quantas artes existiam por classificação para cada país, conseguiria mostrar brevemente as artes, criando assim uma visualização muito rica de informação, mostrando de forma muito interessante a cultura, de forma bem geral, dos principais países de nossa base de dados. A implementação da treemap foi algo relativamente complexo, pois a estrutura normal para se criar um treemap seria um json, assim já deixando clara as hierarquias, fazer isso com um csv não é tão trivial, mas é legal dar esse ar um pouco mais complexo e o resultado fica bem mais legal que outras visualizações mais simples que passaram por nossa cabeça.

2) *Implementação*: Para conseguir mostrar de maneira clara e interessante a cultura diversos países, foram utilizadas diversas interatividades, vamos passar e falar um pouco do

motivo de uma a uma. Como o nosso objetivo era a análise individual por país, a primeira interação com o usuário, instintivamente, é o dropdown que permite o filtro de país, alterando todo o gráfico, pegando apenas as imagens que temos com autores com a nacionalidade selecionada. Assim que é selecionada a nacionalidade, o site faz todos os requests necessários para pegar todas as imagens, se trocar de página e voltar para a mesma, todas as imagens serão carregadas quase instantaneamente.

O tooltip é um ponto extremamente importante para essa visualização, ele passa uma quantidade muito grande de informação. Dado que não foi possível colocar título da categoria de forma interessante, dado que algumas categorias ficavam em áreas muito pequenas, a solução para mostrar a classificação de determinada seção do treemap foi a de mostra no tooltip, além de mostrar quantas artes daquela classificação e naquela nacionalidade, estão ali, isso é bem importante, pois a quantidade de imagens que aparece na visualização não é a quantidade de imagens que tem na categoria, isso será comentado mais a baixo. Além desse "título" destacado no tooltip, temos informações da arte em específico, mostra as principais características da arte e mostra a imagem abaixo, com um tamanho maior que a imagem é vista inicialmente. Além de tudo isso, o tooltip foi tratado com o cuidado de não sair da pagina, ele tem um limite para se redimensionar e, com um certo limite, ficar sempre na tela do usuário, resolvendo um problema que existia ao ver as imagens muito na borda.

Também foi implementado a funcionalidade de zoom utilizando o scroll do mouse ou com um click duplo na imagem, além da opção de "arrastar" a treemap clicando nela e movendo o mouse. Essa funcionalidade é muito interessante dado que facilita a visualização das imagens de forma mais clara, como algumas nacionalidades tem um número grande de imagens e várias categorias, com o zoom é possível ver as categorias com menos imagens, que ficam bem pequena. Após alterar a imagem com o zoom e/ou movê-la, o usuário pode voltar ao estado inicial da visualização apertando a tecla "q". Ao apertar "q" em qualquer lugar da página a pessoa começa a ser levada diretamente à treemap, com a visão do dropdown, não importa a tela que ela esteja acessando.

Além disso, foi fortemente considerada a responsividade de página, fazendo o treemap aproveitar sempre o máximo possível da página independente do monitor e também deixando visível o dropdown para que o usuário possa a qualquer momento escolher uma nacionalidade diferente. Em caso de redimensionamento brusco manual após abrir a pagina, basta recarrega-la com as novas dimensões que o gráfico se ajustará.

3) *Absorção de ideias*: Após a apresentação de nosso trabalho em aula, tivemos diversas sugestões e críticas, nessa área vou abordar todos os comentários feitos a respeito da treemap, o que implementamos e o que achamos que não

agregaria tanto. A principal crítica recebida foi ao problema de carregar muitas imagens, foram sugeridas diversas formas para tentar contornar esse problema, entre elas, download de todas as imagens e tela de espera para o usuário enquanto faz todas as requisições, mas a que achamos mais viável e interessante foi a de reduzir a dimensão das imagens, então atualmente todas as nossas imagens têm 420px de altura e de largura e a ideia de samples, agora pegamos no máximo 20 imagens por classificação e nacionalidade, mantendo a proporção como comentado acima, dessa forma a visualização fica extremamente mais fluida e agradável ao usuário.

Outro ponto de crítica foi as cores apresentadas, foi comentado sobre as cores serem mais chamativas que as imagens e também dito para colocar uma legenda ao lado mostrando qual categoria cada cor representava, mas por uma opção estética, escolhemos seguir outra sugestão, a de deixar uma cor neutra de fundo da imagem e adicionar bordas delimitando cada área, e codificando a legenda no tooltip, dessa maneira acreditamos ter chegado em um resultado bem mais estético e informativo de forma geral.

Outra sugestão enviada foi a de centralizar a visualização, mas seguimos a ideia que o professor falou de aproveitar toda a pagina, assim deixando inteiramente responsiva e permitindo aproveitar tudo, dessa forma já ficando "centralizada". Por fim, foi sugerido filtrar mais elementos no treemap, mas acreditamos que para essa página em específico fugiria um pouco do objetivo que é representar as culturas dos países.

#### IV. CONCLUSÃO

Neste trabalho, exploramos o poder da visualização de dados no contexto de um museu, utilizando JavaScript e a biblioteca D3 para criar visualizações interativas envolventes. O objetivo principal foi analisar e comunicar informações relevantes sobre a coleção de arte, proporcionando aos visitantes uma experiência enriquecedora e significativa. Ao longo do processo, enfrentamos diversos desafios, desde a obtenção e pré-processamento dos dados até a implementação das interações e aprimoramento estético das visualizações. No entanto, esses obstáculos nos permitiram adquirir conhecimentos valiosos sobre as melhores práticas de visualização de dados e explorar as capacidades do JavaScript e do D3.

Por meio das visualizações interativas, conseguimos gerar todo uma experiência ao usuário, permitindo ver as artes em diferentes contextos, ver mais sobre a história de artistas e até mesmo ver de forma geral a cultura de diferentes lugares do mundo.

Embora tenhamos obtido resultados positivos, reconhecemos que sempre há espaço para melhorias. Uma possível área de aprimoramento seria a implementação de recursos adicionais, como a inclusão de informações contextuais e a integração com outras mídias, como áudio e vídeo. Além disso, o teste e a avaliação contínuos com os visitantes do museu seriam cruciais para validar e aprimorar a eficácia das visualizações.

Em conclusão, este trabalho demonstrou o poder da visualização de dados como uma ferramenta eficaz para comunicar informações complexas de forma clara e atraente. As visualizações interativas em JavaScript e D3 trouxeram vida à coleção de arte do Carnegie Museum of Art's, permitindo aos visitantes explorar, descobrir e apreciar de forma mais profunda o patrimônio artístico. Esperamos que este trabalho inspire futuras pesquisas e contribua para a evolução da visualização de dados no campo dos museus e das artes.

#### REFERENCES

- [1] Carnegie Museum of Art, Carnegie Institute, <https://carnegieart.org>, June 2023.
- [2] David Newbury, ., davbre, ., Zac Yu, ., Tracey Berg-Fulton, ., Neil Kulas, ., & Matthew Lincoln, . (2015). collection: Multi-file JSONs (0.3.0). Zenodo. 10.5281/zenodo.35013
- [3] Fernando Serboncini 2021, Substrate, acessado 16 06 2023 <<https://canvas.rocks/substrate/>>
- [4] D3 treemap documentation, <https://d3-graph-gallery.com/treemap.html>
- [5] Api Wikipedia, [https://pt.wikipedia.org/wiki/Wikip%C3%A9dia:Central\\_de\\_pesquisas/Portal\\_de\\_dados/API](https://pt.wikipedia.org/wiki/Wikip%C3%A9dia:Central_de_pesquisas/Portal_de_dados/API)