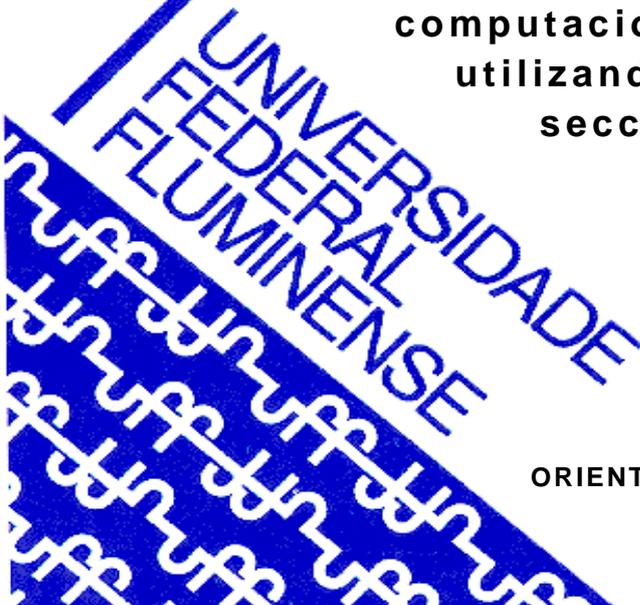




UNIVERSIDADE FEDERAL FLUMINENSE
CENTRO TECNOLÓGICO
ESCOLA DE ENGENHARIA INDUSTRIAL METALÚRGICA DE VOLTA REDONDA
PÓS-GRADUAÇÃO EM ENGENHARIA METALÚRGICA

DISSERTAÇÃO DE MESTRADO

**Aplicação e validação de um código
computacional para reconstrução 3D
utilizando imagens oriundas do
seccionamento em série**



**ALUNO: DOUGLAS SILVA GOMES
ORIENTADOR: WESLEY LUIZ DA SILVA ASSIS**

**VOLTA REDONDA
2021**

UNIVERSIDADE FEDERAL FLUMINENSE
ESCOLA DE ENGENHARIA INDUSTRIAL METALÚRGICA DE VOLTA REDONDA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA METALÚRGICA

Douglas Silva Gomes

**Aplicação e validação de um código computacional
para reconstrução 3D utilizando imagens oriundas
do seccionamento em série**

Dissertação apresentada ao Programa de Pós-graduação em Engenharia Metalúrgica da Universidade Federal Fluminense, como requisito parcial para obtenção do título de Mestre em Engenharia Metalúrgica.

Área de Concentração: Modelamento e Simulação de Processos e Fenômenos.

Orientador: Wesley Luiz da Silva Assis, D.Sc.

**Volta Redonda
2021**

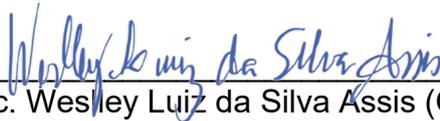
Douglas Silva Gomes

**Aplicação e validação de um código
computacional para reconstrução 3D utilizando
imagens oriundas do seccionamento em série**

Dissertação apresentada ao Programa de Pós-graduação em Engenharia Metalúrgica da Universidade Federal Fluminense, como requisito parcial para obtenção do título de Mestre em Engenharia Metalúrgica.

Área de Concentração: Modelamento e Simulação de Processos e Fenômenos.

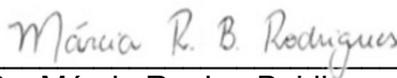
BANCA EXAMINADORA



Prof. D.Sc. Wesley Luiz da Silva Assis (Orientador)
Universidade Federal Fluminense



Prof. D.Sc. Glaucio Soares da Fonseca
Universidade Federal Fluminense



Prof. D.Sc. Márcia Regina Baldissera Rodrigues
Universidade Federal de Itajubá

Volta Redonda
2021

Douglas Silva Gomes

Aplicação e validação de um código computacional para reconstrução 3D utilizando imagens oriundas do seccionamento em série

Dissertação apresentada ao Programa de Pós-graduação em Engenharia Metalúrgica da Universidade Federal Fluminense, como requisito parcial para obtenção do título de Mestre em Engenharia Metalúrgica.

Área de Concentração: Modelamento e Simulação de Processos e Fenômenos.

Ficha catalográfica automática - SDC/BEM
Gerada com informações fornecidas pelo autor

G633a Gomes, Douglas Silva
Aplicação e validação de um código computacional para reconstrução 3D utilizando imagens oriundas do seccionamento em série / Douglas Silva Gomes ; Wesley Luiz da Silva Assis, orientador. Volta Redonda, 2021.
117 f.

Dissertação (mestrado)-Universidade Federal Fluminense, Volta Redonda, 2021.

DOI: <http://dx.doi.org/10.22409/PPGEM.2021.m.12691241700>

1. Seccionamento em série. 2. Código computacional. 3. Reconstrução tridimensional. 4. Produção intelectual. I. Assis, Wesley Luiz da Silva, orientador. II. Universidade Federal Fluminense. Escola de Engenharia Industrial e Metalúrgica de Volta Redonda. III. Título.

CDD -

Bibliotecário responsável: Debora do Nascimento - CRB7/6368

Dedico este trabalho à Deus, minha família e a todos os meus amigos que contribuíram na realização deste sonho.

AGRADECIMENTOS

Primeiramente gostaria de agradecer a Deus pela força proporcionada durante todo o período do curso de Mestrado em Engenharia Metalúrgica e por realizar mais um sonho em minha vida, pois sem Ele isso nunca seria possível.

Gostaria, em especial, de agradecer a minha esposa Ludmila pelo companheirismo e motivação durante toda a trajetória do curso, antes, durante e agora. Nos momentos em que mais precisei ela estava ao meu lado, sempre dando bons conselhos e me ajudando em tudo o que eu precisava direta e indiretamente.

Os agradecimentos ao meu avô Nelson, com sua imensa humildade, por me ensinar as coisas simples da vida e que para ser feliz não precisa de muito, apenas o necessário. Que ele esteja nos braços de Deus, sendo mais uma estrela a iluminar o céu. Tenho certeza Vô, que o senhor está dando muitas risadas aí de cima como sempre fez aqui conosco.

Aos meus familiares, em especial minha mãe Letícia, o meu pai Fernando e meu irmão também Fernando por acreditarem em meu potencial e pela motivação proporcionada durante essa trajetória.

Ao Orientador e professor D.Sc. Wesley Luiz da Silva Assis por me ajudar e orientar sempre que precisei durante o desenvolvimento deste trabalho, a todo o corpo docente e demais funcionários da Universidade Federal Fluminense de Volta Redonda (UFF/VR).

E por fim, porém não menos importante, a todos os meus amigos do trabalho e da UFF pela amizade e companheirismo proporcionado durante o mestrado.

RESUMO

As características intrínsecas de determinadas amostras podem ser identificadas com exatidão a partir de estudos em três dimensões. Um método muito empregado nos estudos da morfologia dos materiais é o seccionamento em série em conjunto com a reconstrução 3D. No trabalho proposto, o processo de seccionamento em série é utilizado de duas formas, sendo uma a forma destrutiva, através de lixamento e polimento, na qual consome toda a amostra durante o procedimento e a outra é a não destrutiva que, com o auxílio da ressonância magnética, é possível fazer o seccionamento completo sem destruir a amostra. O objetivo do trabalho é aplicar e validar um código computacional com finalidade de gerar imagens em 3D de forma fidedigna por meio de imagens em 2D extraídas de um processo de seccionamento em série, seja ele destrutivo ou não. O código foi escrito em linguagem C++ utilizando o Visual Studio em conjunto com o OpenCV. Todavia, neste trabalho, aplica-se um código capaz de reconstruir o domínio discreto em 3D que possibilita visualizar a geometria tridimensional a partir de distintos softwares, tais como: ParaView™ e TecPlot®. A validação do modelo foi feita pelo programa ImageJ, pois o mesmo é utilizado em várias aplicações que podem ser encontradas na literatura. Observou-se que, com a aplicação e validação do código proposto, obteve-se um resultado satisfatório, visto que, a microestrutura reconstruída nos programas propostos ficou conforme o software utilizado para a validação, que neste caso é o ImageJ. Outro ponto importante é que foi possível utilizar o código computacional em imagens reais oriundas de duas diferentes técnicas de seccionamento em série, destrutivas e não destrutivas, comprovando assim que, independente da aplicação de seccionamento que resulte em imagens 2D, o algoritmo é capaz de reconstruir o material com fidelidade.

Palavras-chave: Seccionamento em série, código computacional, reconstrução tridimensional, validação do software.

ABSTRACT

The intrinsic characteristics of attributions can be accurately identified from studies in three dimensions. A method widely used in the studies of the morphology of materials is the serial sectioning together with 3D reconstruction. In the proposed work, the serial sectioning process is used in two ways, one being destructive, through sanding and polishing, in which the entire sample is consumed during the procedure and the other is the non-destructive one that, with the aid of resonance magnetic, it is possible to perform the complete sectioning without destroying the sample. The objective of the work is to apply and validate a computational code with wear and tear to reliably generate 3D images through 2D images extracted from a serial sectioning process, whether destructive or not. The code was written in C++ language using Visual Studio together with OpenCV. The serial sectioning technique is a method already known in the scientific community. However, in this work, a code capable of reconstructing the discrete domain in 3D is applied, which makes it possible to visualize a three-dimensional geometry from different software, such as: ParaView™ and TecPlot®. The model was validated using the ImageJ program, as it is used in several applications that can be found in the literature. It was observed that, with the application and validation of the proposed code, a satisfactory result was obtained, since the microstructure reconstructed in the proposed programs was in accordance with the software used for validation, which in this case is ImageJ. Another important point is that it was possible to use the computational code in real images from two different serial sectioning techniques, destructive and non-destructive, thus proving that, regardless of the sectioning application that results in 2D images, the algorithm is capable of reconstructing the stuff faithfully.

Keywords: Serial sectioning, computational code, three-dimensional reconstruction, software validation.

SUMÁRIO

1. INTRODUÇÃO	14
1.1 Objetivos gerais e específicos	18
1.1.1 Objetivo geral:.....	18
1.1.2 Objetivos Específicos:	18
1.1.3 Exequibilidade do projeto:.....	19
2. REVISÃO DA BIBLIOGRAFIA.....	20
2.1 Conceito de cores	20
2.2 Canais de cor.....	23
2.3 Intensidade da cor.....	24
2.4 Fotografia digital	25
2.5 Imagem digital	26
2.5.1 Processamento de imagens	27
2.6 Função contínua x função discreta	29
2.7 Amostragem e quantização.....	30
2.8 Paradigma dos quatro universos.....	32
2.9 Seccionamento em série de forma não destrutiva.....	34
2.9.1 MicroCT	35
2.10 Seccionamento em série de forma destrutiva	38
2.11 Microscopia Eletrônica de Varredura (MEV).....	41
2.12 Feixe de Íon Focalizado (FIB).....	42
2.13 Micromiller Alkemper-Voorhees	43
2.14 Equipamento <i>Robot-Met.3D</i>	45
2.15 Vantagens e desvantagens do seccionamento em série	46
2.15.1 Vantagens	46
2.15.2 Desvantagens	47
2.16 Binarização das imagens	48
2.17 Tecnologias utilizadas	49
2.17.1 Linguagem de programação C++	49
2.17.2 Microsoft Visual Studio.....	50
2.17.3 OpenCV	51

2.17.4 Ferramentas de visualização.....	52
2.17.5 Ferramenta de validação do algoritmo.....	53
2.18 Reconstrução em 3D da microestrutura.....	53
2.19 Material metálico – ferro fundido.....	55
2.19.1 Tipos de ferro fundido.....	55
3. MATERIAIS E MÉTODOS.....	60
4. RESULTADOS E DISCUSSÃO.....	69
5. CONCLUSÃO.....	85
6. TRABALHOS FUTUROS.....	87
7. REFERÊNCIAS BIBLIOGRÁFICAS.....	88

LISTA DE FIGURAS

Figura 1: O Espectro Visível.....	20
Figura 2: Células sensíveis à luz, Cone e Bastonetes	21
Figura 3: Representação da percepção da cor por um observador	21
Figura 4: Processo aditivo de cores primárias.....	22
Figura 5: Processo subtrativo de cores secundárias	23
Figura 6: Processo de obtenção das cores secundárias.....	23
Figura 7: Subespaço do modelo RGB a partir dos eixos XYZ.....	25
Figura 8: Representação gráfica das informações visual e descritiva.....	28
Figura 9: Representação gráfica de uma função contínua.....	29
Figura 10: Representação Gráfica de uma Função Contínua e Discreta...	30
Figura 11: Imagem de Amostragem e Quantização	31
Figura 12: Representação dos quatro universos	33
Figura 13: Ilustração do equipamento de microCT	37
Figura 14: Esquemático do Seccionamento em Série	39
Figura 15: Seções transversais utilizadas para a reconstrução em 3D	40
Figura 16: Microscópio eletrônico de varredura EVO MA10	42
Figura 17: (a) Equipamentos FIB e MEV acoplados (b) Aparência do Sistema como um todo	43
Figura 18: Imagem do equipamento Micromiller	44
Figura 19: Equipamento RobotMet.3D de bancada:.....	46
Figura 20: Exemplo de binarização	49
Figura 21: Ícone do Microsoft Visual Studio	50
Figura 22: Ícone do <i>OpenCV</i>	51
Figura 23: Ícones dos softwares de visualização.....	52
Figura 24: Ícones do ImageJ	53
Figura 25: Reconstrução 3D após o seccionamento em série.....	54
Figura 26: Ferro fundido branco - a) Microestrutura com zoom de 100x... b) Microestrutura com zoom de 200x	56
Figura 27: Ferro fundido cinzento - a) Microestrutura com zoom de 200x b) Microestrutura com zoom de 500x	57
Figura 28: Ferro fundido mesclado	57
Figura 29: Ferro fundido maleável - a) Microestrutura com zoom de 100x b) Microestrutura com zoom de 200x	58
Figura 30: Ferro fundido nodular - a) Microestrutura com zoom de 100x.. b) Microestrutura com zoom de 200x	58
Figura 31: Ferro fundido vermicular - a) Microestrutura com zoom de 500x b) Microestrutura com zoom de 10000x	59
Figura 32 : Passo a passo da metodologia empregada.....	61
Figura 33: Imagem do Robo-Met.....	63
Figura 34: Imagem do Phoenix V-tomex-m.....	65

Figura 35: Tela inicial de inserção de dados	70
Figura 36: Console 1 de depuração Visual Studio	70
Figura 37: Console 2 de depuração Visual Studio	71
Figura 38: Console 3 de depuração Visual Studio	72
Figura 39: Reconstrução tridimensional - Tecplot	72
Figura 40: Reconstrução tridimensional - ParaView	73
Figura 41: Carregamento das 200 imagens	73
Figura 42: Geração do volume – ferro fundido	74
Figura 43: Volume gerado pelo ImageJ – ferro fundido	75
Figura 44: Ferro fundido – TecPlot, Paraview e ImageJ	75
Figura 45: Nódulo de ferro fundido – TecPlot, Paraview e ImageJ	76
Figura 46: Aba de inserção de dados	77
Figura 47: Console 4 de depuração Visual Studio	78
Figura 48: Console 5 de depuração Visual Studio	79
Figura 49: Console 4 de depuração Visual Studio	79
Figura 50: Reconstrução 3D do paralelepípedo de alumínio envolvido em resina - TecPlot	80
Figura 51: Reconstrução 3D do paralelepípedo de alumínio envolvido em resina - ParaView	80
Figura 52: Carregamento das 201 imagens	81
Figura 53: Geração do volume – paralelepípedo de alumínio envolvido em resina	82
Figura 54: Volume gerado no ImageJ (parte 1) – paralelepípedo de alumínio envolvido em resina	83
Figura 55: Volume gerado no ImageJ (parte 2) – paralelepípedo de alumínio envolvido em resina	83
Figura 56: Paralelepípedo de alumínio – TecPlot, ParaView, ImageJ e imagem real	84
Figura 57: Abrindo o arquivo com a extensão .dat	92
Figura 58: Diretório do arquivo	93
Figura 59: Primeira visualização do arquivo .dat	93
Figura 60: Ativando a caixa “ <i>Contour</i> ”	94
Figura 61: Ativando a caixa “ <i>Contour</i> ”	94
Figura 62: Rotacionando a microestrutura	95
Figura 63: Opções de visualização	95
Figura 64: Opções de visualização	96
Figura 65: Abrindo o arquivo no ParaView 1	97
Figura 66: Abrindo o arquivo no ParaView 2	97
Figura 67: Visualização do arquivo	98
Figura 68: Arquivo impresso na tela – visão de topo	99
Figura 69: Arquivo impresso na tela – visão tridimensional	99
Figura 70: Utilização de ferramentas de rotação	100
Figura 71: Opções de análise	100
Figura 72: Volume detalhado	101

Figura 73: Ferramentas de contorno	101
Figura 74: Arquivo com utilização da ferramenta de contorno	102
Figura 75: Opção para salvar a microestrutura em arquivo de imagem .	102
Figura 76: Abrindo o arquivo no ParaView	103
Figura 77: Arquivo de imagem	103
Figura 78: Importando as imagens em sequencia	104
Figura 79: Diretório das imagens.....	105
Figura 80: Carregamento das imagens bidimensionais	105
Figura 81: Geração do volume	106
Figura 82: Volume criado.....	106
Figura 83: Volume com escala de cores <i>Thermal LUT</i>	107
Figura 84: Salvando a imagem tridimensional	107
Figura 85: Tipos de extensão para a geração de imagens	108

LISTA DE QUADROS

Quadro 1: Informações do computador pessoal.....	68
---	-----------

LISTA DE SIMBOLOS E ABREVIATURAS

RGB	Vermelho, Verde e Azul
CMY	Ciano, Magenta e Amarelo
CMYK	Ciano, Magenta, Amarelo e Preto
2D	2 Dimensões
3D	3 Dimensões
MEV	Microscopia Eletrônica de Varredura
FIB	Feixe de Íons Focalizado
LVDT	Transmissor Diferencial de Variação Linear
POO	Programação Orientada a Objetos
ANSI	<i>American National Standards Institute</i>
IDE	Ambiente de Desenvolvimento Integrado
MicroCT	Microtomografia Computadorizada
CT	Tomografia Computadorizada

1. INTRODUÇÃO

O estudo das formas geométricas de objetos a partir de imagens é antigo tanto quanto a fotografia. O primeiro nome dado a esse tipo de aplicação foi conhecido como fotogrametria. A fotogrametria é basicamente a obtenção das distâncias e dimensões reais dos objetos de estudos por meio da fotografia (FARIAS, 2012).

O presente trabalho teve como motivação a dificuldade em desenvolver algoritmos capazes de recriar domínios computacionais com fidelidade a realidade. Domínios computacionais, conhecidos também como volumes de controle, são regiões onde possibilitam a realização de cálculos específicos de variáveis que modificam-se dinamicamente e, na maioria das vezes, podem ser modelados a partir do desenvolvimento de malhas computacionais, com representação por meio da computação gráfica, equações algébricas entre outros (PRAZERES; ASSIS; GRAÇA, 2018). Com o avanço da tecnologia na área de computação, ferramentas de visualização foram desenvolvidas com o objetivo de representar modelos físicos através de ferramentas computacionais, permitindo que a simulação computacional seja feita de forma fidedigna, se aproximando cada vez mais do mundo real. Destaca-se que a representação de domínios computacionais, em casos onde as geometrias de estudo são complexas, a análise com fidelidade a simulação computacional é colocada em risco devido à complexidade das geometrias.

A representação fiel da microestrutura via simulação computacional é de extrema importância, pois cada milímetro diferente da forma real pode fazer a diferença para alcançar bons resultados, podendo acarretar em interpretações não condizentes com a realidade, comprometendo todo o estudo.

Um exemplo, de acordo com (GRAÇA, 2019) é a análise de tumores em regiões complexas do corpo humano. Com o advento da tecnologia, o surgimento de novas técnicas trouxe novas formas de armazenar arquivos de qualquer natureza, como por exemplo, que são objetos de estudos do presente trabalho, as fotografias. A fotografia surgiu em meados do século XIX, se tornando uma tecnologia revolucionária da época na área das artes visuais (OLIVEIRA, 2006). A fotografia, na sua forma analógica, foi uma tecnologia que

possibilitou a perduração de uma cena ao longo dos anos. Essa tecnologia foi criada em 1826 por Joseph Nicéphore Niépce, no entanto, em 1975 um engenheiro da Kodak criou a primeira câmera digital, que só começou a ser comercializada na década de 90, possibilitando o armazenamento dos arquivos gerados em computadores, disquetes, CD-Rom ou cartões de memória nos formatos que são conhecidos atualmente (OLIVEIRA, 2006).

A microscopia de uma forma geral é uma técnica que fornece imagens bidimensionais (2D) de uma determinada microestrutura sem os detalhes quantitativos diretos sobre a estrutura interna em três dimensões de um determinado material. O conceito de microestrutura pode ser entendido como uma estrutura detalhada de uma determinada amostra, obtendo-a por meio de um microscópio. Cada microestrutura demonstra características únicas de um determinado material, tais como o tamanho dos grãos, a forma geométrica, morfologia entre outros (ZANKEL; WAGNER; POELT, 2014). Como existem diferentes tipos de materiais, bem como várias aplicações destes, e geralmente esses materiais têm características tridimensionais, as imagens em duas dimensões (2D) não são suficientes para analisar as características das microestruturas de forma detalhada, sendo assim, na área da metalurgia existem técnicas de modelagem numérica que possibilitam a obtenção desses materiais em três dimensões utilizando a reconstrução 3D via simulação computacional. Para possibilitar uma reconstrução 3D de materiais destinados a área da metalurgia através da simulação computacional, é necessário aplicar um método conhecido como seccionamento em série no material estudado (TEWARI; GOKHALE, 2000).

O seccionamento em série consiste basicamente em obter várias camadas através de cortes seriados (PRAZERES; ASSIS; GRAÇA, 2018), sendo que a cada secção é capturada uma fotografia da seção 2D da microestrutura em um determinado ponto a partir de um referencial no terceiro eixo em relação a imagem bidimensional, possibilitando o armazenamento dessa imagem da amostra no computador. Após a obtenção da imagem do plano da microestrutura da amostra, novas imagens são obtidas até o término do *looping* ou *scanning* durante o procedimento de seccionamento. Esse processo é realizado repetidas vezes, sendo que, após a obtenção das imagens em 2D, é possível tratar e depois renderizar as imagens, através de

um algoritmo específico, possibilitando a visualização da microestrutura em 3D através da simulação computacional, utilizando ferramentas gráficas (ASSIS et al., 2019).

O método de seccionamento em série é um processo muito importante para a obtenção da reconstrução microestrutural, visto que, através deste é possível obter informações da morfologia do material analisado com maior precisão (CERQUEIRA et al., 2015). Ressalta-se que para o presente trabalho, foram utilizadas imagens oriundas de dois tipos de seccionamento em série: destrutivo e não destrutivo. Esses métodos, bem como suas diferenciações, serão abordados nos capítulos seguintes.

Após o seccionamento em série, é necessário fazer a reconstrução 3D do material. De acordo com (ASSIS et al., 2019) a técnica de reconstrução 3D de uma microestrutura via simulação computacional possibilita a visualização da interconectividade, a morfologia, o tamanho e as distribuições espaciais de grãos e precipitados de qualquer parte da reconstrução simulada, facilitando a análise de qualquer área de interesse da microestrutura estudada (LI et al., 1999).

Após a captura das imagens, elas são trabalhadas para compor a reconstrução, dessa forma, cada fotografia é transformada em uma matriz bidimensional e com um algoritmo específico, essas imagens são combinadas, ou seja, sobrepostas entre si formando uma matriz tridimensional, conseqüentemente possibilitando a reconstrução 3D via simulação computacional.

O paradigma da Programação Orientada a Objetos (POO) foi utilizado para desenvolver o algoritmo que possibilita a reconstrução 3D. Pode-se dizer que os objetos são partes reutilizáveis de um programa que o modela. Percebe-se que o uso de uma abordagem de implementação de códigos orientados a objetos tende a tornar as pessoas que trabalham com o desenvolvimento de software mais produtivas do que as que utilizam somente técnicas de programação estruturada. A orientação a objetos permite o desenvolvimento de programas mais fáceis de compreender, corrigir e modificar (DEITEL, P. ; DEITEL, H., 1983).

O presente trabalho possibilitou aplicar e validar um código computacional serial com finalidade de gerar imagens em 3D de forma

fidedigna por meio de imagens em 2D extraídas de um processo de seccionamento em série, seja ele destrutivo ou não, permitindo assim o desenvolvimento do modelamento de domínios discretos. O código foi escrito em linguagem C++ através do software Visual Studio em conjunto com as bibliotecas do OpenCV.

1.1 Objetivos gerais e específicos

1.1.1 Objetivo geral:

O objetivo geral do trabalho é aplicar e validar um código computacional com finalidade de gerar imagens em 3D de forma fidedigna por meio de imagens em 2D extraídas de um processo de seccionamento em série, seja ele destrutivo ou não destrutivo.

1.1.2 Objetivos Específicos:

- ✓ Aplicar e validar um código computacional para fazer a leitura e interpretação de pixels em uma imagem 2D, bem como a reconstrução 3D;
- ✓ Obter e tratar o pacote de imagens utilizadas na entrada de dados inicial;
- ✓ Visualizar a reconstrução tridimensional através de ferramentas gráficas tais como: Tecplot e ParaView;
- ✓ Analisar os resultados obtidos da simulação desenvolvida após a reconstrução em 3D;
- ✓ Validar o código utilizando o ImageJ, pois este é um software conhecido na literatura;
- ✓ Salvar os dados do domínio discreto de forma que possam ser trabalhadas e/ou modificadas após a reconstrução em 3D.

1.1.3 Exequibilidade do projeto:

O trabalho proposto torna-se técnica e economicamente viável, pois a parte de simulação e reconstrução 3D foi totalmente desenvolvida com recurso próprio e também da infraestrutura existente na Escola de Engenharia Industrial Metalúrgica de Volta Redonda (EEIMVR-UFF), no Núcleo de Modelamento Microestrutural da UFF (NMM).

As imagens de ferro fundido, oriundas do seccionamento em série de forma destrutiva, foram disponibilizadas pelo pesquisador Ph.D. Veeraraghavan Sundar, proporcionando uma parceria internacional para o desenvolvimento da dissertação.

As imagens de paralelepípedo de alumínio envolvido em material de resina, oriundas do seccionamento em série de forma não destrutiva, foram disponibilizadas pelo professor M.Sc. Wyslân Jefferson Lima Garção, aluno de doutorado da UFRJ – Universidade Federal do Rio de Janeiro, proporcionando uma sinergia entre instituições.

2. REVISÃO DA BIBLIOGRAFIA

2.1 Conceito de cores

A formação da cor do sistema visual humano é a partir de impulsos elétricos gerados através de sensores sensíveis à luz, que existem nos olhos. A emissão desses diferentes impulsos elétricos é feita através desses sensores que enviam esses sinais elétricos ao cérebro, variando a cor primária de acordo com o comprimento da onda. Após a recepção desses impulsos elétricos pelo cérebro, este por sua vez, processa esses sinais e resulta na sensação da visão das cores. As cores que são visualizadas pelo sistema visual humano são oriundas de combinações de cores primárias, dessa forma, o cérebro divide as cores em regiões conforme a figura1 (LEITE, 2006).

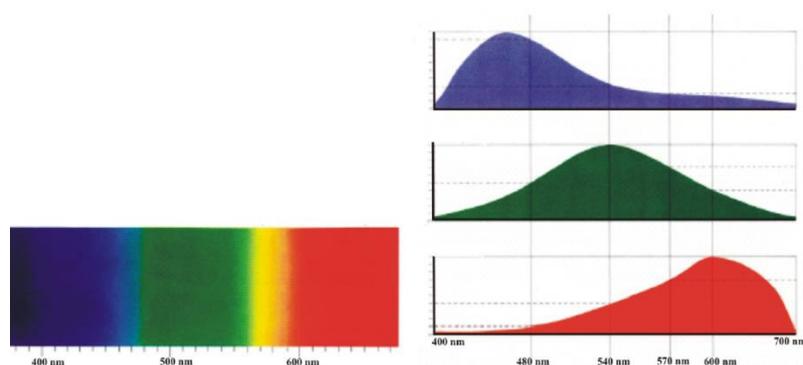


Figura 1: O Espectro Visível
Fonte: (LEITE, 2006)

Como dito anteriormente os olhos humanos possuem elementos fotossensíveis que convertem a imagem visual em impulsos elétricos, esses elementos são denominados como bastonetes e cones. Os bastonetes são responsáveis por detectar as imagens em preto e branco, sendo que, por sua vez, os cones são responsáveis por detectar as cores. Os cones possuem 3 tipos de fotopigmentos fundamentais, cone S (curto), cone M (médio) e cone L (longo) (RIBEIRO, 2011). Os cones e bastonetes podem ser visualizados na figura 2.

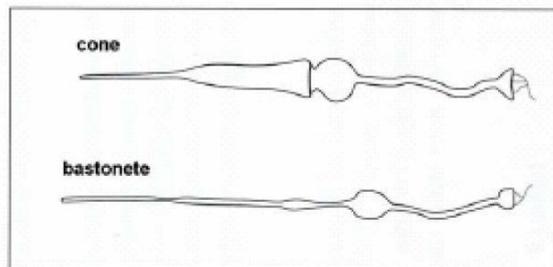


Figura 2: Células sensíveis à luz, Cone e Bastonetes
Fonte: (LEITE, 2006)

As cores não são características predominantes de um determinado objeto, mas sim uma questão de percepção humana. Pode-se dizer que cada indivíduo tem a sua própria percepção de cor em função da quantidade de luz que é irradiada no objeto observado (RIBEIRO, 2011).

Alguns objetos como a brasa e o sol são ditos luminosos, pois irradiam luz. Mas, a grande maioria dos objetos que podem ser vistos a olho nu não são luminosos. O que possibilita a visão dos objetos pelos olhos humanos é a luz que refletem nesses objetos (GRAÇA, 2019). Observa-se, na figura 3, uma representação esquemática da percepção e o julgamento da cor por um indivíduo.

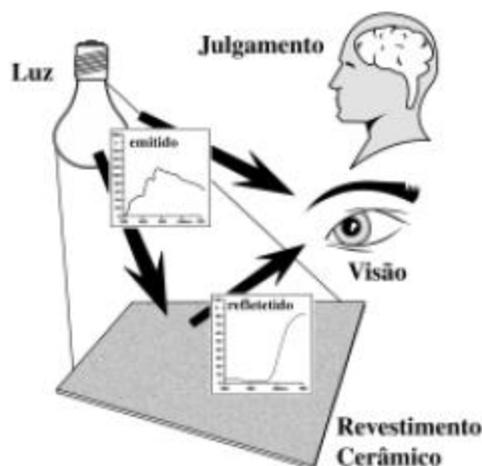


Figura 3: Representação da percepção da cor por um observador
Fonte: Adaptada de (MELCHIADES; BOSCHI, 1999)

Em várias áreas, é necessário trabalhar com cores em diferentes situações, sejam elas para configurar as tonalidades de cores da imagem no Photoshop, Paint, Corel Draw ou até mesmo de um navegador de internet, como por exemplo, o Internet Explorer. Atualmente, encontra-se as cores em

fotografias, televisões, monitores de computador, filmes, celulares entre outros (LEITE, 2006). A formação das cores são compostas basicamente por 3 cores primárias: vermelho, verde e azul que traduzidas para o inglês são: *red*, *green* e *blue*, formando a sigla RGB e também por suas cores complementares ditas secundárias conhecidas como ciano, magenta e amarelo (CMY) do inglês *cyan*, *magenta* e *yellow* (LEÃO; ARAÚJO; SOUZA, 2014).

De acordo com (ROCHA, 2010), existem dois sistemas de cores que são descritos como aditivos e subtrativos. No processo aditivo de cores, o preto indica a ausência de cores e o branco é a soma de todas elas. Um exemplo de utilização desse processo são os televisores ou monitores de computador em geral. Observa-se na figura 4 um sistema RGB de cores.

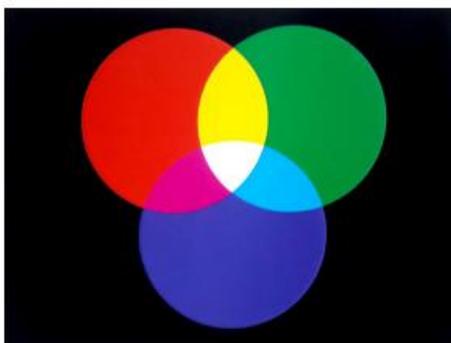


Figura 4: Processo aditivo de cores primárias
Fonte: (AZEVEDO; CONCI, 2003)

A formação de um pixel, em imagens coloridas, é obtida de acordo com o nível de intensidade das três cores primárias em conjunto. O pixel é capaz de variar entre essas cores primárias desde o preto até o branco. A representação da cor C pode ser descrita matematicamente através da equação (2.1):

$$C = r.R + g.G + b.B \quad (2.1)$$

Onde R, G e B são as três cores primárias e r, g e b são os coeficientes de mistura correspondentes aos níveis de R, G e B separadamente. Dessa forma, a cor C de cada pixel da imagem relacionada pode ser representada no espaço tridimensional através das coordenadas de r, g e b que são os coeficientes de mistura citados anteriormente.

No processo subtrativo de cores, o branco corresponde à ausência de cores e o preto é a presença de todas elas. Este processo é dito como o inverso do RGB. Um exemplo de utilização desse processo são as impressoras. Observa-se na figura 5 um sistema CMYK (*cyan, magenta, yellow e black*) de cores.

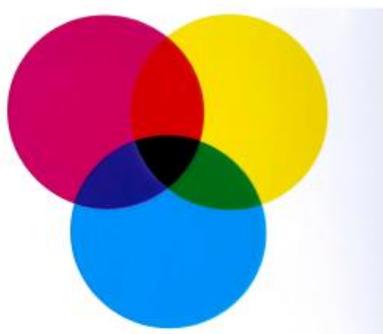


Figura 5: Processo subtrativo de cores secundárias
Fonte: (AZEVEDO; CONCI, 2003)

Como dito anteriormente, é possível obter outras cores através da combinação das cores primárias RGB (*red, green e blue*) e secundária CMYK (*cyan, magenta, yellow e black*). A título de exemplo, seguem algumas combinações de cores na figura 6.

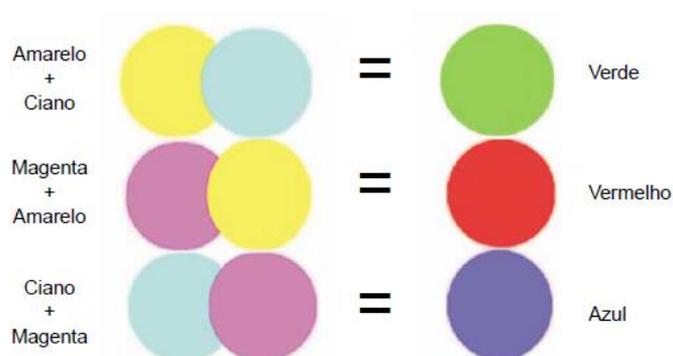


Figura 6: Processo de obtenção das cores secundárias
Fonte: (AZEVEDO; CONCI, 2003)

2.2 Canais de cor

Cada uma das três cores primárias (RGB) de uma imagem pode ser controlada de forma separada em um computador, sendo conhecida como um canal de cor. Admitindo-se que em um canal são utilizados 8 bits por pixel, há a possibilidade de combinar os três canais existentes RGB – vermelho, verde e

azul, tendo 24 bits por pixel. Ressalta-se que uma imagem com apenas tons de cinza, ou seja, preto e branco, só tem um canal de cor, dessa forma, ocupa menos espaço de armazenamento (BUENO, 2010).

2.3 Intensidade da cor

De acordo com (BUENO, 2010), a intensidade da cor pode ser chamada de profundidade de pixel, profundidade de cor ou até mesmo profundidade de bit. Sendo assim, a intensidade da cor é a capacidade que a imagem tem de representar as cores em diferentes tonalidades com variações pequenas ou grandes. A variação se modifica em função da intensidade, sendo que elas são diretamente proporcionais, dessa forma, quanto maior a intensidade maior será a sua variação.

Quando uma imagem tem 8 bits por canal, essa por sua vez possui (2^8) cores, ou seja, 256 cores por canal. Salienta-se que existem três cores primárias (RGB), assim, é possível obter 24 bits por pixel (3 canais x 8 bits).

A partir desse conceito, é possível obter cerca de 16,7 milhões de cores diferentes através da discretização das imagens em 256 níveis no canal de R (vermelho), 256 níveis no canal de G (verde) e 256 níveis no canal de B (azul) (GRAÇA, 2019), sendo que, essa quantidade de tons pode ser projetada nos monitores de computadores atuais e trabalhadas de acordo com a necessidade exigida. A título de curiosidade, em tarefas que requerem altíssimas qualidades e gradações sutis de cores, com maiores precisões, é necessário trabalhar com profundidade de cor maior, como 36 bits, ou seja, 16 bits por canal de RGB, dessa forma, possibilita a formação de uma imagem com 4096 cores por canal, gerando um total de aproximadamente 68,7 bilhões de cores distintas (TEWARI; GOKHALE, 2000).

De acordo com (AZEVEDO; CONCI, 2003), o modelo RGB possui o vermelho, verde e azul como cores primárias e também é dito como cores aditivas. Este modelo se baseia na sensibilidade do olho, utilizando um sistema de coordenadas cartesianas. Observa-se, na figura 7, o subespaço de interesse como um cubo unitário.

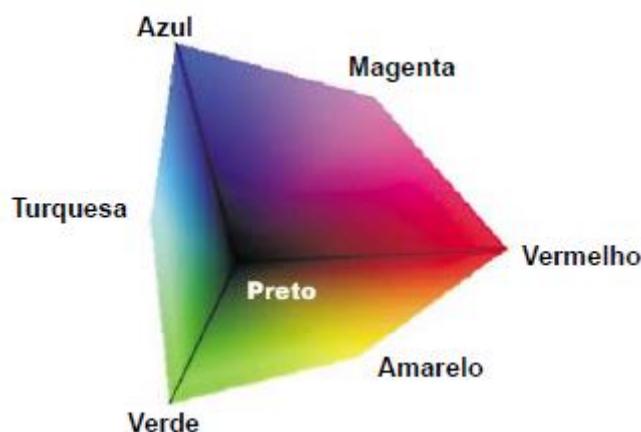


Figura 7: Subespaço do modelo RGB a partir dos eixos XYZ
Fonte: (AZEVEDO; CONCI, 2003)

A diagonal principal do cubo vai do preto ao branco, possuindo quantidades idênticas das cores primárias (vermelho, verde e azul) (AZEVEDO; CONCI, 2003). Em trabalhos computacionais, geralmente aplica-se o processo de discretização (amostragem e quantização).

Salienta-se que o sistema escolhido para implementar as técnicas do algoritmo proposto é o RGB devido ao uso da biblioteca OpenCV, de processamento de imagens, utilizado como parte integrante deste trabalho. Ressalta-se também que existem vários sistemas de cores que não foram descritos nesta dissertação, mas, eles podem ser encontrados na referência (AZEVEDO; CONCI, 2003). Como leitura complementar, sugerem-se as referências (LEITE, 2006), (ROCHA, 2010) e (AZEVEDO; CONCI, 2003).

2.4 Fotografia digital

A fotografia surgiu em meados do século XIX, sendo uma tecnologia revolucionária da época na área das artes visuais (OLIVEIRA, 2006). A observação dos eclipses solares por astrônomos e físicos da época através de câmeras obscuras, princípio básico da máquina fotográfica, propiciou a evolução da fotografia ao longo dos anos. A fotografia, na sua forma analógica, foi uma tecnologia que possibilitou a perduração de uma cena ao longo dos anos. Essa tecnologia foi criada em 1826 por Joseph Nicéphore Niépce, no entanto, em 1975 um engenheiro da Kodak criou a primeira câmera digital, que só começou a ser comercializada na década de 90, possibilitando o armazenamento dos arquivos gerados em computadores, disquetes, CD-Rom

ou cartões de memória nos formatos que são conhecidos atualmente (OLIVEIRA, 2006).

Os arquivos digitais, gerados a partir das fotografias capturadas, possuem algumas informações importantes, tais como: tamanho da imagem, número de pixels, níveis de cores em cada pixel, formato da imagem, quantidade de bytes para armazenamento dentre outros. Além de todas essas informações, o mais importante é a forma como os pixels são organizados, formando o sentido sintático da fotografia em questão. Existem basicamente três níveis de forma e conteúdo, nos quais, são eles:

Sintaxe: É a organização visual dos elementos da imagem;

Morfologia: É a forma da fotografia;

Semântica: É o significado da imagem.

A organização visual dos elementos da imagem é responsável por posicionar cada pixel no seu devido lugar, sendo assim, a representação de formas geométricas complexas, por exemplo, são mantidas a rigor e podem ser representadas de forma real (SCURI, 1999).

2.5 Imagem digital

De uma forma digital, é possível descrever uma imagem como um plano onde há pontos, no entanto, observa-se que no mundo real existem infinitos pontos, mas, de acordo com (GRAÇA, 2019) uma imagem está relacionada como um par de números reais que aponta para outro número real, ou seja, representando de forma simbólica, esse conceito pode ser entendido como $R \times R \rightarrow R$, sendo que, um par de números reais (x,y) pode ser considerado como um ponto do plano da imagem e o outro número real, no qual está sendo apontado pelos outros dois números reais, pode ser considerado como a cor desejada. Dessa forma, a cor pode ser representada por números, mas, sabe-se que os computadores não trabalham com números reais, apenas com combinações de zeros e uns. Os computadores em geral são capazes de contar os pontos de uma imagem, visto que, no mundo físico esse processo é impossível. Mas para que o computador possa contar esses pontos é

necessário submetê-los a um processo denominado como Amostragem, que nada mais é, que a seleção de alguns pontos do mundo físico, independente das quantidades, desde que seja enumeráveis, e depois passar esses pontos para um computador de modo que seja possível contá-los, ou seja, como dito anteriormente, pontos enumeráveis. O computador em si não é capaz de representar todas as cores sozinho, dessa forma, para contar as cores em um computador, lembrando que as cores também são contadas, necessita-se de outro processo que se denomina como quantização, ou seja, é um processo no qual há a discretização do espaço de cor de uma determinada imagem no espaço (GRAÇA, 2019).

2.5.1 Processamento de imagens

Uma imagem digital possui basicamente dois tipos de informações: A Visual e a Descritiva. Informação visual é aquela que pode ser vista na tela de um computador, celular, televisores entre outros. Informação descritiva é aquela que contém o modelo matemático no qual pode representar os objetos visualizados (SCURI, 1999). Na computação gráfica existem três áreas nas quais são descritas abaixo a função de cada uma delas:

Processamento de imagens: São operações realizadas sobre imagens e que resultam em imagens;

Computação gráfica: São operações que abrangem a síntese da imagem, ou seja, a geração de uma visualização do modelo construído;

Visão computacional: São operações que analisam os objetos contidos na imagem digital e também fazem a geração de modelos matemáticos desses objetos.

Observa-se, na figura 8, de forma ilustrativa, as três áreas comentadas anteriormente.

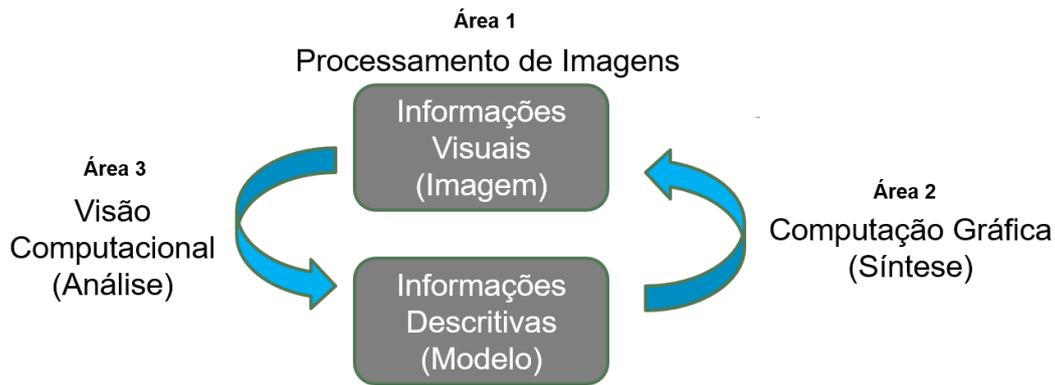


Figura 8: Representação gráfica das informações visual e descritiva
Fonte: Adaptada de (SCURI, 1999)

De acordo com (SCURI, 1999), em processamento de imagens utiliza-se a forma matricial como modelo e em computação gráfica geralmente trabalha utilizando o modelo de objetos vetoriais. Existem diversas aplicações para os dois modelos, nos quais podem ser exemplificados abaixo:

Exemplo de utilização do modelo matricial: Análise da porosidade de um material metálico. Neste caso, necessita-se de uma fotografia digital para que seja possível analisar a amostra em questão.

Exemplo de utilização do modelo vetorial: Projeto de encaminhamento de circuitos elétricos (desenvolvido no software AutoCAD), utilizando eletrodutos para passagem dos cabos elétricos. Neste caso não há a necessidade de uma fotografia, visto que é possível obter o desenho a partir das suas medidas.

Dessa forma, essas diferenças de modelos podem levar a diferentes formas de trabalhos. Sabendo-se que, em processamento de imagens é utilizado o modelo matricial, que por sua vez, trabalha como uma matriz de dados na qual é armazenada a informação de cor em cada ponto da imagem, conseqüentemente, exigindo mais armazenamento em memória. Já a computação gráfica utiliza-se o modelo vetorial, no qual, os objetos são armazenados apenas a partir da descrição das coordenadas dos seus respectivos vértices, sejam eles com duas ou três dimensões. Sendo assim, utiliza-se um sistema de coordenadas cartesiano, onde esses objetos podem ser rotacionados, transladados ou até mesmo escalados. Salienta-se que o modelo vetorial exige menos memória do que o modelo matricial.

Para mais detalhes sobre Processamento de Imagens e Computação Gráfica, recomenda-se a leitura na referência (SCURI, 1999).

2.6 Função contínua x função discreta

Matematicamente pode-se dizer que uma função contínua é aquela na qual para cada valor de x sempre existe um único valor de $f(x)$, sendo que, uma função contínua em seu domínio deve ser contínua em todos os pontos que pertencem ao seu domínio, por outro lado, quando a função não atende esse conceito, ela é dita como descontínua ou uma função discreta (SCURI, 1999). Portanto dados contínuos são dados quantitativos que podem tomar todos os valores numéricos que estão compreendidos no seu intervalo de variação, já os dados discretos são os dados quantitativos que só podem ter um número, seja ele finito ou infinito, desde que seja enumerável, com valores distintos (GRAÇA, 2019). A título de exemplo, pode-se imaginar uma função contínua sendo uma corda sem emendas que pode se estender infinitamente, e mesmo cortando essa corda, essa por sua vez, poderá sempre ser cortada de forma infinitesimal (SCURI, 1999).

Observa-se graficamente na figura 9 o exemplo demonstrado anteriormente.

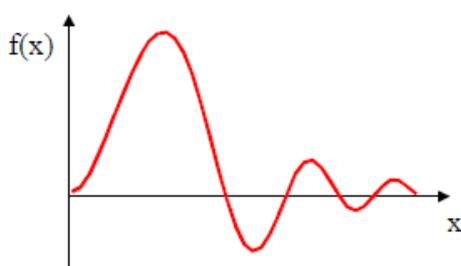


Figura 9: Representação gráfica de uma função contínua
Fonte: (SCURI, 1999)

A operação que traz uma função contínua para o computador é conhecida como discretização ou digitalização, sendo que, para que seja possível fazer esse processo é necessário ter os valores pontuais ao longo do eixo x , conhecido como domínio e atribuí-los ao eixo $f(x)$, conhecido como contradomínio. Lembrando que ambos os eixos de x e $f(x)$ devem ser discretizados, pois eles são contínuos.

De acordo com (GRAÇA, 2019), a discretização do domínio, ou seja, do eixo x , é conhecida como um processo de Amostragem e a discretização do contradomínio, ou seja, do eixo $f(x)$, é conhecida como um processo de Quantização.

Observa-se, na figura 10 os gráficos das funções contínua e discreta.

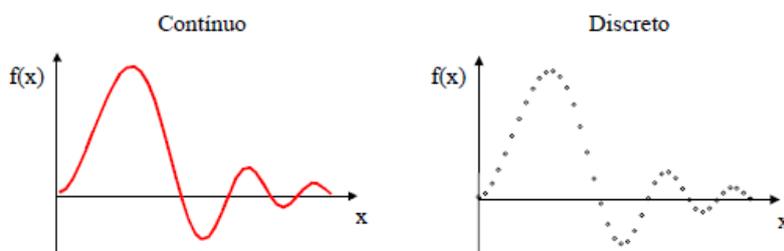


Figura 10: Representação Gráfica de uma Função Contínua e Discreta
Fonte: Adaptada de (SCURI, 1999)

2.7 Amostragem e quantização

Uma imagem pode ser processada de forma digital desde que ela tenha a sua função $f(x,y)$ digitalizada tanto nas coordenadas espaciais quanto na amplitude. Entende-se como digitalização das coordenadas espaciais como amostragem e digitalização da amplitude de quantização (GONZALES; WOODS, 2010).

De acordo com (ASSIS et al., 2019), a amostragem pode ser entendida como um processo de digitalização ao longo dos eixos horizontal e vertical da imagem, por outro lado, a quantização pode ser entendida como um processo de digitalização dos valores da amplitude da imagem. Pode-se dizer que os processos de amostragem e quantização estão ligados a resolução da imagem e são diretamente proporcionais entre si, sendo que, quanto maior o número de pontos a serem amostrados na imagem, maior será a resolução desta mesma imagem. Da mesma forma que, quanto maior o número de níveis de intensidade luminosa da imagem, no qual representa a tonalidade do ponto amostrado, maior será a resolução desta imagem. Ainda de acordo com (ASSIS et al., 2019), a organização de uma imagem pode ser representada por matrizes, onde o número de *pixels* das linhas e colunas da matriz é determinado pela amostragem, e o valor numérico dos elementos da matriz pode ser extraído pelo computador através do processo de quantização. Pode-se verificar, na figura 11, o processo de amostragem e quantização.

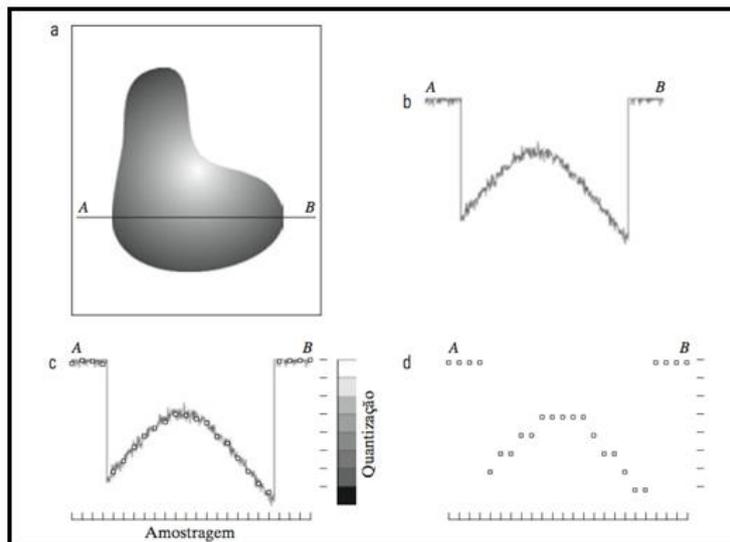


Figura 11: Imagem de Amostragem e Quantização
Fonte: Adaptada de (GONZALES; WOODS, 2010)

Observa-se que na figura 11, a imagem (a) se refere a uma imagem contínua f que se deseja converter em um formato digital. Na figura (b) são observados os valores de amplitude da imagem contínua ao longo do segmento da reta AB mostrada na imagem (a). Após a definição dos valores numéricos da amplitude da imagem destacados na imagem (b) através de níveis de cinza, a imagem (c) mostra o processo de amostragem, no qual são definidas as amostras espaçadas de forma igual ao longo do seguimento AB, sendo que, a posição de cada amostra no espaço pode ser indicada por linhas verticais no eixo x do gráfico.

Na imagem (c), verifica-se que as localizações discretas da imagem nos dão a função de amostragem, mas os valores dessas amostras nesse momento ainda preenchem uma faixa contínua dos valores de intensidade da figura 11, ou seja, em níveis de cinza.

Salientando que, para formar uma função discreta é necessário converter os valores de intensidades da imagem, ou seja, fazer a quantização. Observa-se na figura (c) uma escala do lado direito e na vertical com oito níveis de intensidade, variando do branco até o preto. Entende-se que esses níveis indicam valores específicos atribuídos a cada um dos oito níveis de intensidade. A intensidade contínua do nível de cinza é quantizada através da atribuição de um dos oitos níveis discretos de cinza a cada amostra, sendo que, a atribuição é feita de acordo com a proximidade da amostra na marca da

escala vertical da figura (c). Observa-se, na figura (d), que as amostras na forma digital são resultantes do processo de amostragem e da quantização. De acordo com (GONZALES; WOODS, 2010), o processo de amostragem e quantização é feito linha por linha iniciando na parte superior da imagem e por consequência gerando uma imagem digital em duas dimensões. Para o detalhamento do processo de amostragem e quantização, recomendam-se as referências (SCURI, 1999) e (GONZALES; WOODS, 2010).

2.8 Paradigma dos quatro universos

Através do paradigma dos quatro universos é possível entender, de forma detalhada, a conversão de uma imagem do mundo físico, no qual os dados são contínuos, para o mundo da computação, onde os dados são discretos. Basicamente o paradigma aponta quatro universos distintos, sendo eles:

Universo físico: Encontram-se nesse universo os dados do mundo físico, ou seja, os objetos ou fenômenos de grandezas analógicas não digitalizadas. Um exemplo desse universo é a voz humana ou dados estatísticos.

Universo matemático: É o universo que possibilita a representação abstrata, através de fórmulas matemáticas que equacionam as grandezas analógicas dos objetos do universo físico, ou seja, do mundo real. Um exemplo desse universo são modelos numéricos.

Universo de representação: É descrito como um universo que permite a inserção do universo matemático, ou seja, descrições abstratas, no mundo digital. Nesse universo são implementados os processos de amostragem e quantização. Dessa forma, é nesse universo que ocorre a discretização dos objetos contínuos.

Universo de implementação: Entende-se como universo de implementação a codificação do sinal discretizado na memória do computador através de uma estrutura de dados definida e também dos algoritmos associados ao universo de representação, sendo possível a utilização de ferramentas para visualização em 3D.

Observa-se, na figura 12, uma representação dos quatro universos.

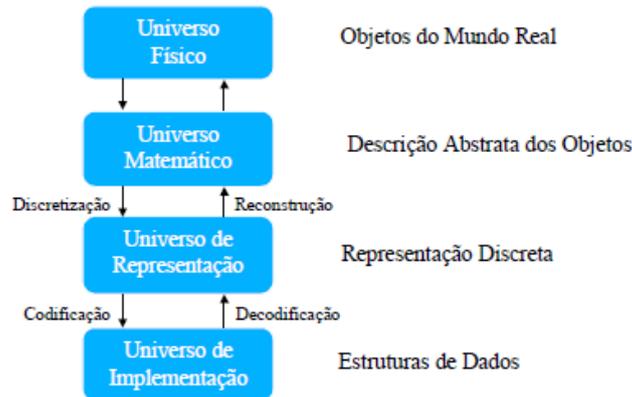


Figura 12: Representação dos quatro universos
Fonte: (SCURI, 1999)

Analogamente, observa-se também, como exemplo, uma bola de cristal com formato esférico, sendo que essa pode ser enquadrada no paradigma dos quatro universos.

- ✓ Universo Físico: Considere os objetos do mundo real sendo uma superfície esférica representada pela bola de cristal;
- ✓ Universo Matemático: A descrição da função do modelo matemático é representada pelas equações (2.2) para a representação implícita e as equações (2.3), (2.4) e (2.5) para a representação paramétrica da esfera em R^3 :

1) Representação implícita (ANDRADE, [s.d.]):

$$(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 = r^2 \quad (2.2)$$

Sendo que “r” é o raio da esfera.

2) Representação paramétrica (ANDRADE, [s.d.]):

$$x = x_0 + r \cos(\theta) \sin(\varphi) \quad (2.3)$$

$$y = y_0 + r \sin(\theta) \sin(\varphi) \quad (2.4)$$

$$z = z_0 + r \cos(\varphi) \quad (2.5)$$

Sendo que: $(0 \leq \theta \leq 2\pi$ e $0 \leq \varphi \leq \pi)$

- ✓ Universo Representativo: A representação são os parâmetros inseridos de acordo com a descrição funcional da bola de cristal exemplificada acima. Neste exemplo citado, pode-se utilizar o nome e raio (n, r).
- ✓ Universo de Implementação: Para fazer a implementação é necessário utilizar a estrutura de dados. Utiliza-se uma associação para descrever a quantidade de objetos, quando necessário, nesse caso obtém-se: $n_k \rightarrow r_k$, sendo que:
 - $k = 1, \dots, Z$, onde Z é a quantidade, quando necessário, que se deseja representar. Nesse caso as bolas de cristal.

2.9 Seccionamento em série de forma não destrutiva

Uma das técnicas de seccionamento em série não destrutiva em materiais é a utilização da microtomografia computadorizada, porém, antes da microCT veio o emprego da tomografia computadorizada amplamente utilizada na medicina atualmente.

A técnica de tomografia computadorizada foi inicialmente utilizada para aplicações clínicas no início da década de 70. Muito utilizado na área médica, o equipamento consiste basicamente em uma fonte de raios X que percorre por um tubo, sendo essa acionada em conjunto com a realização de movimentos circulares, emitindo um feixe de raios X em forma de leque e também contém detectores do lado oposto a fonte que é responsável pela conversão da radiação em sinal elétrico e posteriormente em imagem digital. A sincronia desses elementos possibilita a varredura do leque de feixes em todo o corpo estudado, conseguindo assim, extrair os dados necessários para a análise em forma de imagens (AMARO JÚNIOR; YAMASHITA, 2001).

Como dito anteriormente, a tomografia computadorizada utiliza feixes de raios X que são absorvidos de diferentes formas, a depender da característica de cada material utilizado para a análise. Os elementos considerados como pesados, como por exemplo o cálcio e bário, tem a capacidade de absorver mais feixes de raio X em comparação a elementos leves, tais como os gases hidrogênio e oxigênio ou até mesmo o carbono. Esse fenômeno de distinção entre elementos leves e pesados é conhecido como atenuação, que ocorre

devido à absorção e ao espalhamento do feixe de raio X. O coeficiente de atenuação, propriedade de cada material, é o elemento responsável pela capacidade de atenuar um feixe de raio X, que no caso da tomografia computadorizada atual, utiliza-se um tubo de raios X que possibilita a emissão de um feixe em forma de leque. As informações obtidas pelo equipamento possibilitam a aquisição de imagens bidimensionais que refletem as propriedades do material, tendo como distinção de atenuação escalas de cinzas contidas na própria imagem que é uma reconstrução bidimensional, representada como unidade de área o pixel, que tem tamanhos definidos a depender do equipamento utilizado. Os tons de cinza mais claros representam as atenuações maiores e os tons mais escuros correspondem as medidas de atenuações menores. Cada pixel de uma determinada imagem corresponde, de acordo com o seu respectivo tom de cinza, a uma média ponderada de medidas de atenuação atribuída a um voxel, no qual representa o volume formado pelo pixel e pela profundidade do corte, ou seja, um *gride* regular em um espaço tridimensional (SILVA, 2004).

2.9.1 MicroCT

A microtomografia computadorizada de raio X é um mecanismo que produz imagens que são capazes de possibilitar uma análise detalhada da estrutura interna do material. Essa técnica, de característica não destrutiva, só foi viabilizada devido a descoberta por Wilhelm Conrad Rontgen no final do ano de 1895 (ARAÚJO, 2020).

As origens da microCT vieram da tomografia computadorizada, sendo a CT amplamente aplicada na medicina. Dessa forma, a necessidade de se obter as características tridimensionais da parte interna dos materiais, impulsionou-se a adaptações da técnica de CT para possibilitar a utilização da microTC. Antigamente, após ampla utilização na área médica, e já com conhecimento mais apurado da técnica, utilizava-se a tomografia computadorizada para inspeção de peças industriais, artefatos da arqueologia e estudos na área da geologia (ARAÚJO, 2020).

O princípio de funcionamento de equipamentos de microCT é basicamente a geração dos raios X quando os elétrons são acelerados em direção ao material através do campo elétrico, sendo que, quando esses

elétrons encontram o material analisado, eles são desacelerados. A interação dos feixes de raios X com intensidade I_0 em relação ao material com coeficiente de atenuação μ e espessura x , ocorre quando uma porção dos fótons é absorvida enquanto a parte remanescente passa pelo material, que de acordo com a equação de Beer-Lambert, a intensidade da radiação aplicada ao material é exponencialmente inversa a sua espessura, conforme equação (2.6).

$$I = I_0 e^{-\mu x} \quad (2.6)$$

Equação da intensidade da radiação aplicada ao material
Fonte: (ARAÚJO, 2020)

Onde:

I = Intensidade da radiação

I_0 = Intensidade inicial dos feixes de raios X

μ = Coeficiente de atenuação

x = Espessura do material

e = Exponencial

Salienta-se que na maioria das aplicações de microCT, as fontes são policromáticas, ou seja, possui uma energia com variação de zero até a energia máxima do sistema, e como dito anteriormente, cada material possui um coeficiente de atenuação, permitindo assim a variação com diferentes densidades. Dessa forma, de acordo com a equação (2.7), os trajetos percorridos pelo feixe de raio X podem ser somados em função de cada acréscimo de espessura do material em questão, permitindo a varredura de todo o sistema analisado.

$$I = \int_0^{E_{máx}} I_0 e^{-\int \mu(x,y,E) dI} dE \quad (2.7)$$

Equação da soma dos trajetos percorridos pelos feixes de raios X
Fonte: (ARAÚJO, 2020).

Onde:

I = Intensidade da radiação

E = Energia máxima

I_0 = Intensidade inicial dos feixes de raios X

μ = Coeficiente de atenuação

x, y = Espessura do material

e = Exponencial

Nos dias atuais, a microtomografia computadorizada é utilizada em diversas áreas do conhecimento, com grande aplicação na morfologia dos materiais, análise de suas estruturas entre outros. De acordo com (ARAÚJO, 2020), uma peculiaridade do equipamento é que o tubo de raio X possui um tamanho reduzido conhecido como microfoco, sendo que, quanto menor o foco, melhor é a resolução das imagens, proporcionando uma qualidade maior nas imagens. Ressalta-se que os equipamentos de microCT possuem resolução na ordem de micrômetros, permitindo assim uma melhor qualidade nas análises das estruturas dos materiais, sejam elas internas ou externas. De forma ilustrativa, a figura 13, demonstra basicamente o funcionamento do equipamento.

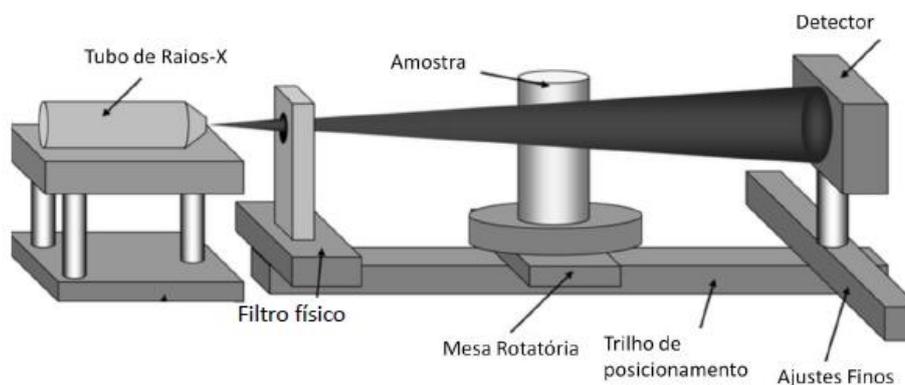


Figura 13: Ilustração do equipamento de microCT
Fonte: (ARAÚJO, 2020)

Inicialmente, de acordo com o tipo de material analisado, o tubo de raio X opera em função da variação de tensão e corrente, gerando assim os feixes de raios X (TELES, 2016). Após os ajustes necessários, esses feixes passam por um filtro físico para atenuar possíveis ruídos inerentes ao sistema. O equipamento possui uma mesa rotatória para que seja possível rotacionar o material em 360° (ARAÚJO, 2020), pois é necessário adquirir vários ângulos a fim de se obter várias projeções do material, sendo que a cada projeção obtida no sistema, melhora cada vez mais a relação entre sinal e ruído na detecção pelos sensores (TELES, 2016).

O equipamento possui um trilho de posicionamento e outro trilho para ajustes finos, sendo estes na direção x e y da máquina, a fim de se estabelecer o melhor local e alinhamento para análise dependendo do tipo e tamanho do material. E por último o equipamento possui um detector com sensores capazes de absorver os feixes de raios X e possibilitar que o sistema faça a conversão para sinais digitais e posteriormente adquirir as imagens da peça em análise (TELES, 2016).

2.10 Seccionamento em série de forma destrutiva

De acordo com (CERQUEIRA et al., 2015), desde o surgimento do ensaio de investigação da metalografia em 1863, as informações de visualizações obtidas das imagens oriundas de seções transversais de metais propiciou para o desenvolvimento da engenharia metalúrgica. A partir desse contexto, segundo (CERQUEIRA et al., 2015), Hillert no ano de 1962 desenvolveu uma técnica que possibilitava a investigação de uma colônia de perlita através do corte serial da microestrutura. No artigo relacionado, ele produziu cerca de 242 seções metalográficas através do seccionamento seriado e ainda produziu um filme em movimento para poder entender a morfologia tridimensional da perlita.

O seccionamento em série consiste basicamente em obter várias camadas através de cortes seriados utilizando o processo de lixamento e polimento cíclico de uma amostra. Destaca-se que o primeiro processo é o lixamento, no qual é, geralmente, utilizada uma lixa grossa que possibilite a retirada da camada do objeto submetido ao procedimento. Já o segundo processo é o polimento, onde, geralmente, utiliza um pano fino ou um material que seja capaz de possibilitar brilho à superfície (VOORHEES, 2001). A amostra é lixada por etapas em uma determinada direção e após esse passo ela recebe o polimento. Após o término desses dois passos, de lixamento e polimento, uma fotografia da superfície da microestrutura é capturada, possibilitando o armazenamento dessa imagem da amostra no computador. Após a obtenção da imagem do plano da microestrutura da amostra em questão, novas imagens são obtidas após o *loop* do procedimento de seccionamento. Conforme citado anteriormente, o procedimento é realizado

repetidas vezes, sendo que, após a obtenção das imagens em 2D, oriundas do processo de seccionamento, é possível renderizar as imagens, através de um algoritmo específico, possibilitando a visualização da microestrutura em 3D através da simulação computacional (ASSIS et al., 2019).

O método de seccionamento em série é um processo muito importante para a obtenção da reconstrução microestrutural, visto que, através deste é possível obter informações da morfologia do material analisado com maior precisão (CERQUEIRA et al., 2015).

De forma didática, observa-se na figura 14 um fluxograma do processo de seccionamento, de forma destrutiva, descrito anteriormente.

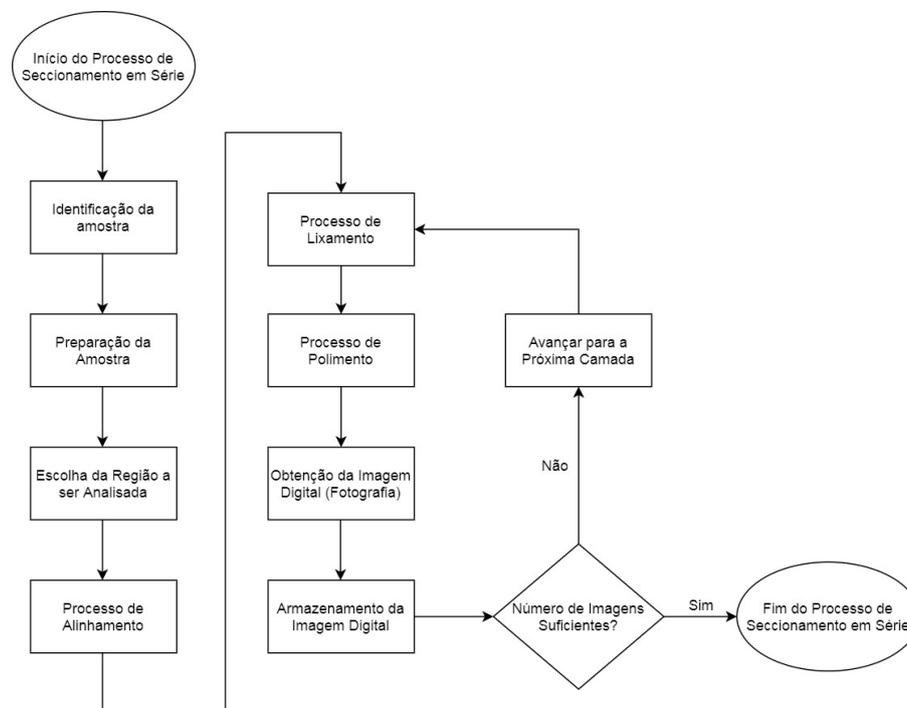


Figura 14: Esquemático do Seccionamento em Série
Fonte: Autoria Própria (2021)

Ressalta-se que o método de seccionamento em série descrito acima é uma técnica utilizada de forma manual, que, segundo a literatura, era muito utilizada antigamente. Porém, atualmente existem equipamentos capazes de executar essas tarefas de forma automática, sem a intervenção humana. Pode-se observar, a partir dos capítulos seguintes, uma descrição detalhada de alguns equipamentos utilizados para a técnica de seccionamento em série.

De forma ilustrativa, segundo (UCHIC et al., 2012), a figura 15 mostra as seções transversais de uma amostra com forma esférica, ou seja, os cortes

realizados em um objeto com forma cúbica esférica através do processo de seccionamento em série. Ainda de acordo com (UCHIC et al., 2012), são necessárias de 50 a 150 seções transversais, em média, para a captura de informações do objeto. Destaca-se que existem três problemas que devem ser considerados, pois estes podem prejudicar a obtenção da reconstrução em 3D. De forma sucinta, destacam-se abaixo esses problemas:

Distância: É de suma importância garantir que as distâncias entre as camadas sejam conhecidas com exatidão.

Alinhamento: É necessário garantir que as imagens estejam alinhadas em relação ao seu deslocamento.

Inclinação Horizontal: A inclinação horizontal deve ser conhecida e todas as seções devem ter o mesmo ângulo de inclinação.

Observa-se na figura 15 o exemplo de cortes seriados de uma forma esférica conforme citado anteriormente.

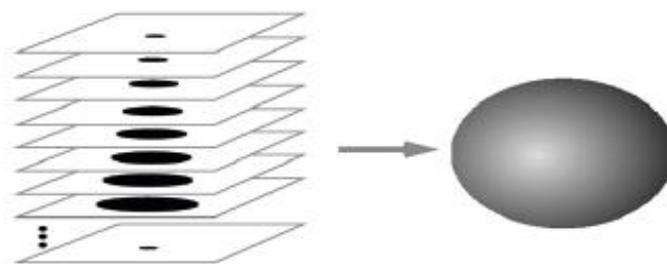


Figura 15: Seções transversais utilizadas para a reconstrução em 3D
Fonte: (UCHIC et al., 2012)

Após a explanação dos principais problemas da técnica de seccionamento em série, ressalta-se que, quanto maior a distância entre as camadas, maior é a perda de informação para a reconstrução 3D da microestrutura. Porém, quanto menor a distância entre as camadas, maior será o trabalho experimental para obter as seções.

Além dos três problemas citados anteriormente, outro fator que deve ser considerado durante a técnica de seccionamento em série é a espessura a ser utilizada para retirada da amostra em estudo, pois essa espessura define um meio termo entre a quantidade de camadas e a distância entre seções. De acordo com (ASSIS et al., 2019) o desbastamento da amostra gera significativas perdas de informações, sendo assim, para amenizar esse problema, aplicam-se métodos computacionais para determinar a

microestrutura entre dois planos discretos e também métodos que representem de forma real as perdas de informações durante o processo de desbaste da amostra.

Salienta-se que a obtenção das fotografias geralmente é feita através da microscopia óptica, dessa forma, como dito anteriormente, o alinhamento é um fator crucial para uma correta reconstrução, sendo assim, é um processo que deve ter atenção para que a análise seja feita corretamente e com clareza (LIEBERMAN, 2007).

Com o avanço da tecnologia, os processos de seccionamento em série foram melhorados, dessa forma, os avanços consideráveis propuseram para que o seccionamento serial pudesse ser combinado com o escaneamento da superfície 2D utilizando técnicas de microscopia adequadas, melhorando cada vez mais as análises das imagens oriundas do processo (GRAÇA, 2019). Assim, destacam-se abaixo dois métodos bastante utilizados no processo de metalografia, são eles: MEV e FIB.

2.11 Microscopia Eletrônica de Varredura (MEV)

Historicamente, a principal função de qualquer microscópio é possibilitar a visualização de objetos que não são possíveis a serem observados a olho nu. De acordo com (DEDAVID; GOMES; MACHADO, 2007), a lupa é o instrumento mais antigo para visualização de pequenos objetos. Logo após, veio o microscópio óptico, no qual ilumina o objeto com uma luz visível ou ultravioleta, possibilitando visualizar amostras com um aumento de aproximadamente milhares de vezes.

Com os avanços tecnológicos, o microscópio eletrônico de varredura, popularmente conhecido como MEV, possibilitou o aumento cada vez mais da resolução para observar uma microestrutura. O MEV é um dos instrumentos mais utilizados atualmente para observação e análise de características microestruturais de materiais sólidos. O primeiro trabalho reconhecido, no qual foi descrito o conceito de um MEV é o do Knoll em 1935 (DEDAVID; GOMES; MACHADO, 2007). O princípio básico de funcionamento de um microscópio eletrônico de varredura (MEV) consiste em utilizar feixes de elétrons de diâmetros pequenos, que são guiados por um sistema de bobinas de deflexão,

passando ponto a ponto pela superfície da amostra emitindo o sinal para uma tela catódica, sendo que a varredura deve estar em perfeita sincronia com o feixe de luz incidente (DEDAVID; GOMES; MACHADO, 2007).

A título de ilustração, a figura 16 mostra o microscópio eletrônico de varredura (MEV), instalado no laboratório de microscopia eletrônica da EEIMVR da Universidade Federal Fluminense.



Figura 16: Microscópio eletrônico de varredura EVO MA10
Fonte: Autoria própria (2021)

2.12 Feixe de Íon Focalizado (FIB)

O Feixe de Íon Focalizado (FIB), oriundo do inglês *Focused Ion Beam*, é uma técnica similar ao MEV (SILVA, 2014), no entanto há uma pequena diferença entre eles, sendo que, enquanto o microscópio eletrônico de varredura (MEV) utiliza um feixe focalizado de elétrons para reproduzir as imagens, o (FIB) feixe de íon focalizado, como o próprio nome diz, utiliza um feixe de íons focalizado para gerar as imagens (ZANKEL; WAGNER; POELT, 2014). Dessa forma, para melhorar as análises das imagens e possibilitar um melhor seccionamento em série das amostras, utiliza-se um equipamento que compõe as duas técnicas descritas acima, no qual é conhecido como (FIB/MEV), ou seja, uma junção do feixe de íon focalizado com o microscópio eletrônico de varredura, possibilitando o aproveitamento das principais especialidades de cada técnica.

Observa-se na figura 17 o FIB/MEV em conjunto, bem como a aparência do sistema.

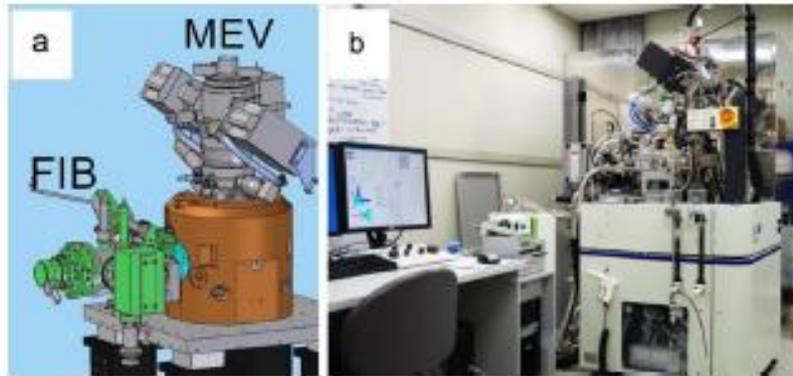


Figura 17: (a) Equipamentos FIB e MEV acoplados (b) Aparência do Sistema como um todo

Fonte: Adaptada de (GRAÇA, 2019) apud (UCHIC et al., 2012)

2.13 Micromiller Alkemper-Voorhees

O equipamento Micromiller é um instrumento que também faz o seccionamento em série e foi desenvolvido na Universidade de *Northwestern* por Alkemper e Voorhees em meados anos 2000 possibilitando uma análise das microestruturas mais automatizada.

Basicamente o seu princípio de funcionamento consiste em utilizar lâminas rotativas de diamante para fazer o corte seriado através do processo de fresagem. A lâmina utilizada como fresa tem a finalidade de preparar a superfície da microestrutura de qualidade óptica em metais, cujas propriedades são dúcteis e ligas que não reagem com o material da lâmina, que no caso é o diamante como o Pb, Sn, Al e ligas de Cu (GHOSH; DIMIDUK, 2011). Apresenta-se na figura 18 o equipamento Micromiller.

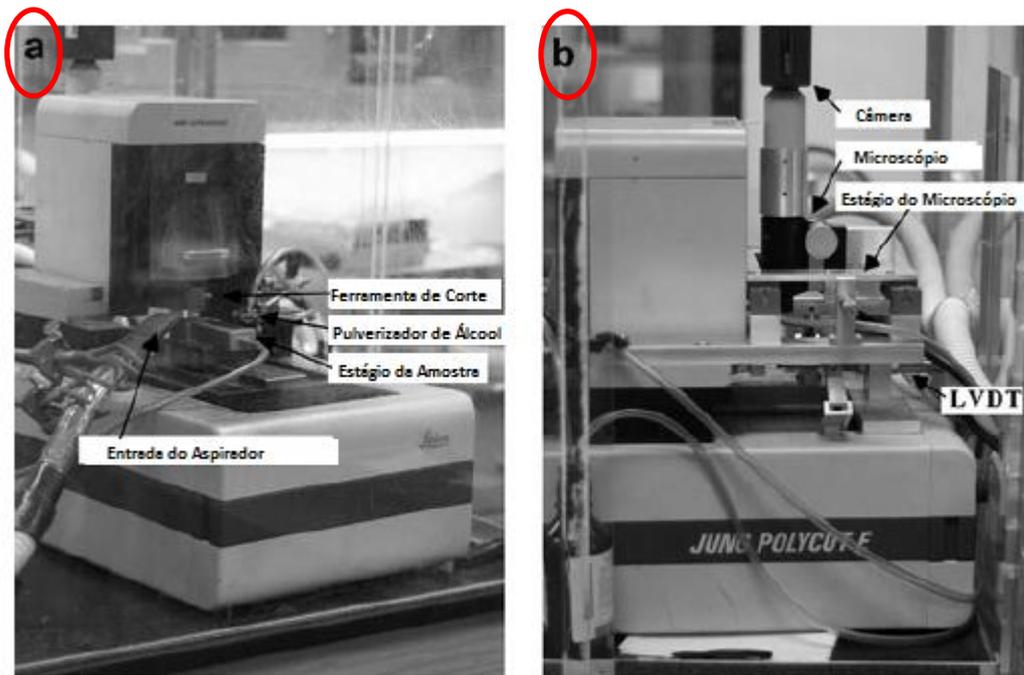


Figura 18: Imagem do equipamento Micromiller
Fonte: Adaptada de (GHOSH; DIMIDUK, 2011)

Observa-se, na figura 18, o dispositivo de seccionamento em série desenvolvido por Alkemper e Voorhees. Salienta-se que na imagem (a) o dispositivo detém de uma ferramenta de corte, um pulverizador de álcool, o estágio da amostra e a entrada do aspirador. Já na imagem (b), observa-se o sistema com uma melhoria, equipando-o com uma câmera, um microscópio e um LVDT (transmissor diferencial de variação linear).

O sistema como um todo possui um micrótomo comercial, que tem a finalidade de fazer os cortes microscópicos, sendo que este micrótomo está equipado com um sistema rotativo responsável pela microfresagem. O equipamento possui também um sistema de lavagem e secagem da amostra durante os cortes seriados, um LVDT oriundo do inglês *Linear Variable Differential Transformer*, que é basicamente um sensor que mede o deslocamento linear em função de uma tensão aplicada nas suas bobinas, e para fazer a captura e gravação da imagem o equipamento dispõe de uma câmera digital com alta resolução que possibilita a captura da imagem após a leitura da superfície da amostra pelo microscópio óptico (GHOSH; DIMIDUK, 2011). Para detalhes do funcionamento do equipamento Micromiller, sugere-se a referência (GHOSH; DIMIDUK, 2011).

2.14 Equipamento *Robot-Met.3D*

O RobotMet.3D é conhecido também como um equipamento que faz o seccionamento em série de amostras (UCHIC et al., 2012). O funcionamento deste dispositivo é semelhante ao Micromiller, pois existem três estações no sistema, que são:

Sistema de corte: Responsável pelos cortes seriados da amostra de estudo;

Sistema de gravação, lavagem e secagem: Responsável pelo tratamento da amostra, camada por camada durante o seccionamento;

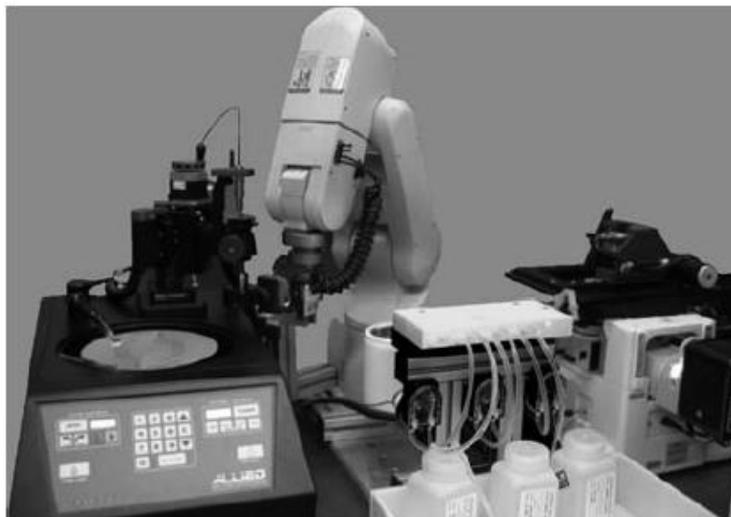
Sistema de captura das imagens: Responsável pela geração e armazenamento das imagens ópticas.

Dessa forma, o Robomet.3D pode ser utilizado para analisar uma variedade de materiais muito maior do que o Micromiller, pois para ele fazer o processo de remoção da amostra, o mesmo utiliza um mecanismo de precisão para o sistema de polimento. Salienta-se que o sistema de polimento da amostra é o método mais usado para preparar metais para a análise metalográfica, e portanto, o sistema de corte utilizado é mais conveniente para examinar vários materiais estruturais (GHOSH; DIMIDUK, 2011).

Cada uma das estações é uma unidade fisicamente separada no RoboMet.3D, sendo que este equipamento possui um braço robótico composto por seis eixos que são destinados a mover a amostra entre as estações e também reter enquanto ela está passando pelo processo de gravação, lavagem e secagem (com ar forçado). Ressalta-se que o microscópio utilizado é um equipamento disponível no mercado, sendo que este é utilizado nesses sistemas de seccionamento em série de forma automatizada, ou seja, o microscópio tem uma automação capaz de executar as tarefas de foco, ajuste de contraste e captura de imagens da superfície analisada sem a interferência humana no processo, acarretando em uma melhor qualidade das imagens geradas (GHOSH; DIMIDUK, 2011).

Por ser um equipamento totalmente automatizado, o Robomet.3D vem ganhando espaço nos processos de seccionamento em série, gerando imagens com qualidade que posteriormente podem ser utilizadas de forma fidedigna na reconstrução 3D.

Observa-se, na figura 19, o equipamento de seccionamento em série automatizado RobotMet.3D.



**Figura 19: Equipamento RobotMet.3D de bancada:
Fonte: Adaptada de (GHOSH; DIMIDUK, 2011)**

Da esquerda para a direita da figura 19, o dispositivo contém: polidor metalográfico de precisão, robô de seis eixos, estação de gravação, lavagem e secagem e um microscópio motorizado invertido.

2.15 Vantagens e desvantagens do seccionamento em série

O seccionamento em série dispõe de certas vantagens e desvantagens, assim como qualquer aplicação de metodologia experimental. Essas características de vantagens e desvantagens podem se modificar de acordo com o tipo de material a ser utilizado durante a execução do procedimento de seccionamento (GHOSH; DIMIDUK, 2011). De acordo com (GRAÇA, 2019), isso ocorre devido às características dos materiais, pois para cada tipo de material, existe uma técnica capaz de gerar imagens bidimensionais em sequência para serem analisadas e/ou reconstruídas através da simulação computacional. Assim, seguem abaixo as vantagens e desvantagens do seccionamento em série de forma convencional.

2.15.1 Vantagens

Uma das vantagens que a implantação da técnica de seccionamento em série possibilita é utilizar apenas uma amostra para o procedimento obtendo várias imagens das micrografias em 2D, com um único custo tanto de material

quanto de equipamento e pessoal (GHOSH; DIMIDUK, 2011). Dessa forma, com a flexibilidade da técnica, o custo final é relativamente baixo, sendo que esses equipamentos podem ser encontrados em laboratórios de materiais, possibilitando uma implantação desse método de uma forma eficaz (GRAÇA, 2019).

Outro fator importante a ser considerado é que a técnica possibilita uma obtenção, a partir das coletas das imagens em 2D, da reconstrução tridimensional da microestrutura via simulação computacional, facilitando uma análise mais detalhada da amostra de estudo, permitindo obter informações sobre as morfologias do material.

2.15.2 Desvantagens

Uma das principais desvantagens do método de seccionamento serial de forma destrutiva é a destruição da amostra de estudo, pois para algumas aplicações é inaceitável, sendo que toda a amostra é consumida durante o processo de coleta dos dados, impossibilitando uma reutilização do material após o procedimento. Por esse motivo é bastante difícil que essa técnica possa competir de forma direta com tecnologias não destrutivas ou não invasivas, como por exemplo, a tomografia computadorizada, que utiliza uma imagem para representar a seção ou um corte de um órgão (UCHIC, 2011). Em 2010 o autor (UCHIC, 2011) mencionou em seu artigo que o processo de seccionamento em série de forma manual é demorado e repetitivo, dessa forma, a automação desses equipamentos reduz significativamente o tédio e o tempo gasto para realizar esse experimento, sendo assim, conseqüentemente, propiciando ganhos significativos na qualidade dos dados que podem ser coletados (DEHOFF, 1983).

Além disso, a automação industrial proporciona outros benefícios que agregam valor para a reconstrução 3D, como por exemplo a variabilidade dos dados que, utilizando uma automação do equipamento, os cortes ficam mais precisos e as espessuras padronizadas, propiciando então uma reconstrução 3D da microestrutura mais confiável.

2.16 Binarização das imagens

Após a obtenção das imagens em duas dimensões (2D) pelo processo de seccionamento em série, necessita-se binarizar as imagens para que posteriormente possa ser feita a reconstrução em três dimensões (3D) da amostra de estudo.

A binarização de imagens é uma técnica simples e bastante eficiente com um olhar para a visão computacional. Em linhas gerais, a binarização é basicamente uma técnica de segmentação ou também conhecida como divisão em segmentos, que consiste em decompor uma imagem nas suas respectivas partes constituintes, ou seja, identificar as partes que são objetos de interesse na imagem e analisá-las posteriormente (ESQUEF, 2002). Essas partes descritas anteriormente podem ser objetos e/ou materiais.

As imagens, oriundas do processamento de corte serial, geralmente são coloridas, ou seja, podem ser representadas no sistema de cores RGB descrito anteriormente. De acordo com (ESQUEF, 2002), para realizar a binarização em uma imagem, primeiramente necessita-se transformar a imagem colorida em uma imagem em tons de cinza. Uma imagem digital, na escala de cinza, pode ter duzentos e cinquenta e seis (256) tons de cinza diferentes, podendo variar do preto com a cor de menor intensidade (0 – zero) e o branco com a cor de maior intensidade (256 – duzentos e cinquenta e seis).

Após a conversão da imagem colorida para uma imagem com tons de cinza, é necessário convertê-la também para uma imagem que tenha representação binária (dois tons), ou seja, preto e branco.

A conversão de uma imagem em níveis de cinza é de suma importância para identificar os objetos e /ou materiais e separá-los do fundo da imagem, analisar a sua geometria (quando esta for mais importante que a intensidade dos pixels) e também possibilitar que a imagem possa ser apresentada em um dispositivo de saída que tem somente dois níveis de intensidade, como por exemplo, uma impressora. Observa-se, na figura 20, um exemplo de binarização.



Figura 20: Exemplo de binarização
Fonte: Adaptada de (GONZALES; WOODS, 2010)

2.17 Tecnologias utilizadas

Os softwares utilizados no presente trabalho servem de base para análise das imagens bidimensionais (2D) oriundas do processo de seccionamento em série, demonstrar a reconstrução tridimensional (3D) através da simulação computacional e possibilitar a visualização dos resultados após a renderização das imagens. Dessa forma, destaca-se a seguir os programas utilizados.

2.17.1 Linguagem de programação C++

Atualmente é possível encontrar diversos softwares desenvolvidos a partir da linguagem de programação C++ em variadas aplicações que nos auxiliam em trabalhos do cotidiano (FELICIANO; LAMEGO, 2008).

Como breve história dessa linguagem, o C++ é uma extensão da linguagem C, no qual esta foi criada e implantada primeiramente por Dennis Ritchie na década de 1970 em um DEC PDP-11 que utilizava o sistema operacional UNIX (SCHILDT, 2000).

Por vários anos, o padrão para C foi à versão desenvolvida com o sistema operacional UNIX, porém, com o advento da tecnologia e a popularidade dos microcomputadores, várias implementações de C foram criadas possibilitando divergências por não existir um padrão da linguagem. Mesmo com essas discrepâncias, por utilizarem o código fonte, estes por sua vez eram altamente compatíveis, possibilitando a compilação. Para corrigir essa situação, o ANSI (*American National Standards Institute*) definiu em um comitê um padrão a ser utilizado para a linguagem C.

Para o desenvolvimento de softwares, a linguagem C permite grande desenvolvimento tecnológico. Destaca-se que sistemas operacionais como *Microsoft Windows*, *GNU-Linux* e *Mac OS-X* foram desenvolvidos em C, bem como para outros tipos de sistemas (FELICIANO; LAMEGO, 2008).

A linguagem C++ é muito utilizada nos dias atuais devido à capacidade de adaptação do código ao problema proposto, permitindo que novas funcionalidades possam ser incorporadas no código e também por possibilitar uma compatibilidade com diversos sistemas operacionais.

Salienta-se que o C++ tem diversas aplicações, como por exemplo, jogos em 3D, pacotes do *Office* (*Word*, *Excel*, *Access*, *coreDRAW*), o navegador de internet *Mozilla* entre outros (FELICIANO; LAMEGO, 2008).

A programação estruturada tem-se mostrado muito eficaz com o passar dos anos, sendo utilizada em larga escala. Destaca-se que uma significativa evolução do C++, que é uma característica do seu desenvolvimento, é a possibilidade da linguagem suportar o emprego do paradigma da Programação Orientada a Objetos – POO, que vem do inglês (*Object Oriented Programming - OOP*), proporcionando a resolução de projetos complexos para programações estruturadas (DEITEL, P. ; DEITEL, H., 1983).

2.17.2 Microsoft Visual Studio

Para a construção do código computacional em C++, utilizou-se o software da Microsoft, conhecido como Microsoft Visual Studio, no qual observa-se, na figura 21 o seu respectivo ícone.



Figura 21: Ícone do Microsoft Visual Studio
Fonte: (VISUAL STUDIO, 2021)

O Microsoft Visual Studio é um ambiente de desenvolvimento integrado (IDE) da Microsoft completo para Android, iOS, Windows, Web e nuvem com vários recursos de programação dedicado para desenvolvimento de software, podendo utilizar as linguagens Visual Basic (VB), C, C++, C#, F#, HTML dentre outros. O software tem muita flexibilidade, possibilitando a criação de

aplicativos para qualquer plataforma, possuindo um depurador onde é possível corrigir os *bugs* rapidamente, permitindo o acesso a variadas extensões e vários outros recursos que auxiliam no desenvolvimento de soluções para diferentes aplicações (VISUAL STUDIO, 2021). Atualmente o Microsoft Visual Studio possui três edições:

Community: Uma edição do IDE (ambiente de desenvolvimento integrado) gratuita orientada para alunos, colaboradores de software livre e desenvolvedores individuais.

Professional: Uma edição do IDE orientada para pequenas equipes de pequenos desenvolvedores.

Enterprise: Uma edição do IDE orientada para soluções escalonáveis de ponta para equipes de qualquer tamanho.

Salienta-se que para a construção do algoritmo para a simulação computacional do presente trabalho, a edição utilizada foi o Community.

2.17.3 OpenCV

Para auxílio no processamento das imagens da reconstrução tridimensional, utilizou-se uma biblioteca totalmente livre conhecida como *OpenCV* (Biblioteca de Visão Computacional de Código Aberto) do inglês Open Source Computer Vision Library, no qual apresenta-se na figura 22 o seu respectivo ícone.



Figura 22: Ícone do *OpenCV*
Fonte: (OPENCV, 2021)

O *OpenCV* foi desenvolvido pela Intel em 1999 por Gary Bradski com o intuito de acelerar as pesquisas e aplicações comerciais de visão computacional pelo mundo e também para a empresa criar uma demanda por computadores cada vez mais poderosos para essas aplicações. Vadim Pisarevsky se uniu a Gary para gerenciar uma equipe de software, mas, com o passar do tempo, a equipe original começou a trabalhar em outras empresas,

consequentemente propagando o conhecimento. Vários membros dessa equipe acabaram trabalhando com robótica e chegaram à Willow Garage. Willow em 2008 viu a necessidade de se avançar rapidamente com os recursos de percepção robótica de forma aberta, alavancando toda a comunidade comercial e de pesquisa, apoiando o OpenCV que, nos dias atuais, nos auxiliam com os processamentos de imagens e vídeos (OPENCV, 2021).

2.17.4 Ferramentas de visualização

Para a visualização da simulação computacional, segue abaixo uma breve descrição dos dois softwares utilizados.



(a) ParaView

(b) Tecplot

Figura 23: Ícones dos softwares de visualização
Fonte: (a) (PARAVIEW, 2021) e (b) (TECPLOT, 2021)

ParaView – É um software de código aberto que possibilita a análise e visualização de dados multiplataforma. Os usuários podem criar visualizações para analisar seus dados rapidamente utilizando técnicas qualitativas e quantitativas. A exploração de dados pode ser realizada de forma interativa em 3D ou através de programação utilizando recursos de processamento em lote do próprio programa. Esse aplicativo foi projetado para analisar conjuntos de dados extensos utilizando recurso de computação em memória distribuída. Outra possibilidade é executá-lo em supercomputadores para analisar conjuntos de dados na ordem de petabytes, bem como em laptops para obter dados menores. Essa ferramenta tornou-se muito útil em laboratórios, universidades e indústrias, ganhando vários prêmios relativos à computação de alto desempenho (PARAVIEW, 2021).

Tecplot – É um software de simulação computacional que auxilia engenheiros e cientistas a analisarem e entenderem informações oriundas de dados complexos. A visualização de dados quando o programa está executando simulações, análises ou experimentos, fornecem um conhecimento e entendimento oculto em seus dados. Essas informações são capazes de

identificar e resolver problemas, otimizar projetos e explicar acontecimentos físicos.

Esse software pode ser trabalhado em conjunto com a linguagem de programação C++, utilizada no presente trabalho, possibilitando a visualização das saídas do algoritmo (TECPLOT, 2021).

2.17.5 Ferramenta de validação do algoritmo

O ImageJ é uma ferramenta de arquitetura aberta capaz de exibir, editar, analisar e processar as imagens possibilitando uma reconstrução tridimensional através do empilhamento das imagens bidimensionais. Esse programa foi escolhido para fazer a validação do código desenvolvido pois o mesmo possui uma vasta aplicação de trabalhos que podem ser encontrados na literatura (IMAGEJ, [s.d.]).

Observa-se na figura 24 o ícone do programa utilizado para a validação do algoritmo proposto.



Figura 24: Ícones do ImageJ
Fonte: Adaptada de (IMAGEJ, [s.d.])

A validação do código foi feita utilizando as mesmas imagens consideradas no código proposto no presente trabalho. Após realizadas as simulações tridimensionais nos dois programas (ImageJ e no código desenvolvido em C++) foi feita uma análise comparativa a fim de confirmar que o algoritmo proposto realmente é capaz de realizar a reconstrução tridimensional da amostra de forma fidedigna ao mundo físico.

2.18 Reconstrução em 3D da microestrutura

Os avanços tecnológicos atrelados à transição da análise da microestrutura em imagens digitais tem manifestado um maior interesse por profissionais que analisam as microestruturas, ou seja, microscopistas, em

programas e equipamentos que sejam capazes de organizar e armazenar grandes quantidades de dados oriundos das imagens digitais através do seccionamento em série (ASSIS et al., 2019). Salienta-se que as imagens digitais com resoluções cada vez maiores possibilitam uma melhor análise e visualização da microestrutura em 3D. De acordo com (ASSIS et al., 2019), a técnica de reconstrução 3D de uma microestrutura via simulação computacional possibilita a visualização da interconectividade, a morfologia, o tamanho e as distribuições espaciais de grãos e precipitados de qualquer parte da reconstrução simulada, facilitando a análise de qualquer área de interesse da microestrutura estudada.

Após a captura das imagens, elas podem ser trabalhadas para compor a reconstrução, dessa forma, cada fotografia é transformada em uma matriz bidimensional e com um algoritmo específico, essas imagens são combinadas, ou seja, sobrepostas entre si formando uma matriz tridimensional, conseqüentemente gerando a reconstrução 3D conforme ilustrado na figura 25.

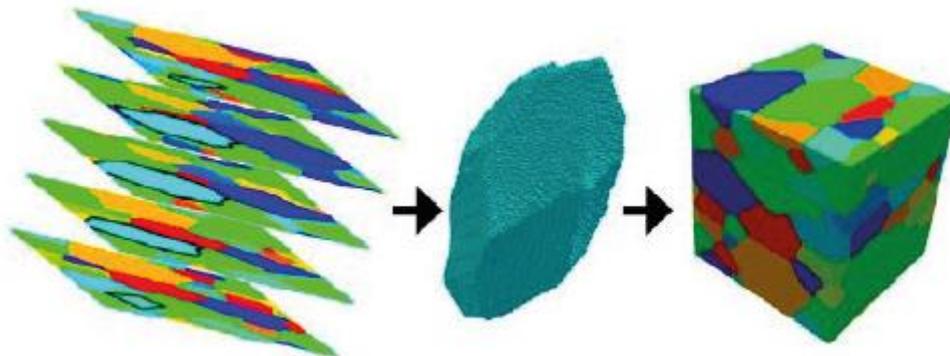


Figura 25: Reconstrução 3D após o seccionamento em série
Fonte: Adaptada de (ASSIS et al., 2019) apud (UCHIC et al., 2012)

Dessa forma, é possível obter informações tridimensionais com detalhes, facilitando uma melhor análise das microestruturas em questão. Destaca-se que o método de seccionamento em série possibilita a obtenção de informações detalhadas da microestrutura após a reconstrução 3D, essa metodologia vem recebendo uma atenção cada vez maior (ASSIS et al., 2019).

Para que a reconstrução 3D seja eficiente, é de suma importância atentar-se ao tratamento das imagens, pois uma falha nesse processo pode comprometer a reconstrução da microestrutura como um todo. As fotografias digitais capturadas podem ser salvas em diferentes formatos de arquivos

digitais. Sabe-se que esses arquivos possuem informações, tais como: resolução, data, hora da captura, número de pixels, níveis de vermelho, verde e azul em cada pixel entre outros. Salienta-se que a correta organização das imagens é um fator crucial para que a fotografia da microestrutura seja mantida a rigor, possibilitando que a representação geométrica da microestrutura via simulação computacional seja fidedigna.

2.19 Material metálico – ferro fundido

O presente trabalho tem um foco maior aos materiais metálicos, em escala microestrutural. Mas, o uso da ferramenta proposta não impede de ser aplicada em outros tipos de materiais como cerâmico ou poliméricos.

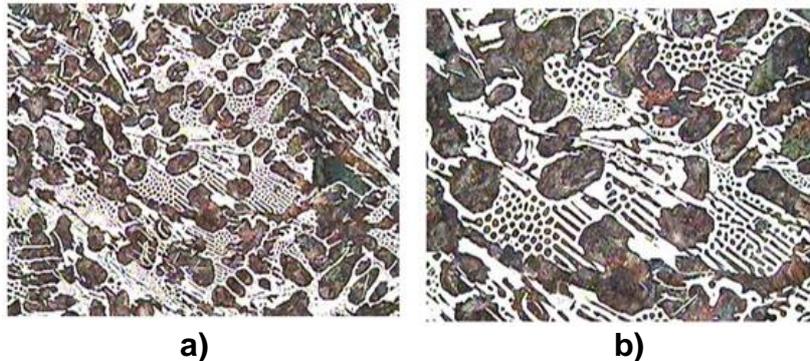
2.19.1 Tipos de ferro fundido

Os ferros fundidos, antigamente, eram materiais com pouca aplicação devido às suas baixas propriedades mecânicas e também pela sua fragilidade. Porém, com os avanços tecnológicos, esse tipo de material se tornou indispensável nos ambientes industriais, possuindo um campo de aplicação diversificado com menores custos de produção e obtendo quantidades satisfatórias de propriedades, possibilitando então o seu uso no cotidiano (CHIAVERINI, 1990).

Basicamente, os ferros fundidos contêm ferro, carbono e silício em sua composição. De acordo com o diagrama de equilíbrio Fe-C o carbono, geralmente aparece com concentrações geralmente em torno de 2%. De acordo com (COSTA, 2003), o ferro fundido pode ser considerado como uma liga ternária Fe-C-Si, sendo que o silício aparece de forma natural juntamente com o carbono possuindo porcentagens similares ou maiores que o carbono. Salienta-se que as propriedades dos ferros fundidos dependem da estrutura de sua matriz e também da forma de precipitação da grafita (GRAÇA, 2019). Segundo (VIDAL, 2013), os ferros fundidos são classificados em seis tipos diferentes, sendo eles:

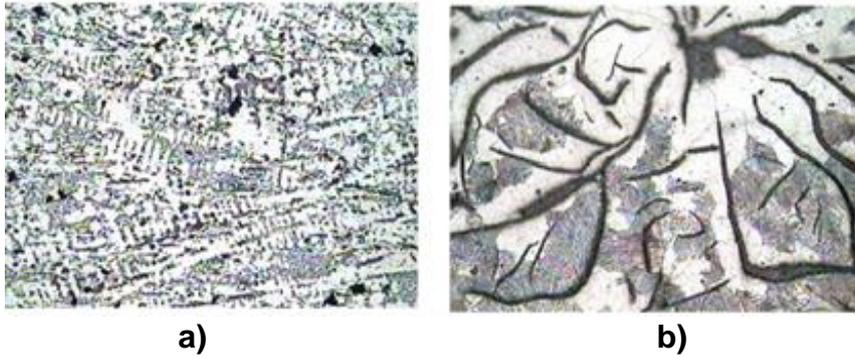
Ferro fundido branco: O carbono se apresenta na forma combinada Fe_3C , possuindo predominância na microestrutura devido à baixa concentração do silício e com a adição de outros elementos como o cromo, no qual é utilizado como estabilizador da cementita. As principais características

presentes são alta dureza e baixa ductilidade, restringindo a sua aplicação industrial. Possui também como característica uma coloração clara nos pontos de fratura, sendo assim conhecido como ferro fundido branco. Esse tipo de ferro fundido é utilizado para matéria prima do ferro fundido maleável, utilizando o tratamento térmico de maleabilização (COSTA, 2003) e (VIDAL, 2013).



**Figura 26: Ferro fundido branco - a) Microestrutura com zoom de 100x
b) Microestrutura com zoom de 200x
Fonte: Adaptada de (SOUZA, 2019)**

Ferro fundido cinzento: O silício se apresenta em alta concentração, proporcionando a formação de carbono livre na forma de veios grafiticos. Neste caso, os principais elementos de liga são o carbono e silício, podendo ter outros como o manganês, cromo e o cobre. A grafita em veios proporciona uma boa condutividade térmica, característica no qual possibilita o seu uso em componentes que são solicitados a fadiga térmica, dessa forma, a sua aplicação é encontrada em tambores e discos de freios ou cabeçotes de motor. Destaca-se também a capacidade do ferro fundido cinzento em atenuar vibrações em energia térmica, característica no qual possibilita o seu uso em bases de máquinas e equipamentos, blocos de motores, carcaças entre outros. Em função da sua coloração escura nas regiões de fratura, esse tipo de ferro fundido é conhecido como “cinzento” (COSTA, 2003) e (VIDAL, 2013).



a) **b)**
Figura 27: Ferro fundido cinzento - a) Microestrutura com zoom de 200x
b) Microestrutura com zoom de 500x
Fonte: Adaptada de (SOUZA, 2019)

Ferro fundido mesclado: É apresentado na microestrutura como uma combinação dos ferros fundidos cinzento e branco em função da parte de sua solidificação ser um sistema estável, ou seja, composto por austenita e grafita e outra parte da sua solidificação ser um sistema metaestável, ou seja, composto por austenita e carbeto. Suas propriedades são intermediárias entre os ferros fundidos branco e cinzento. Sua coloração no ponto de fratura é um brilhante acinzentado em função da mistura dos ferros fundidos branco e cinzento, sendo conhecido por esse motivo, como “mesclado” (COSTA, 2003) e (VIDAL, 2013).

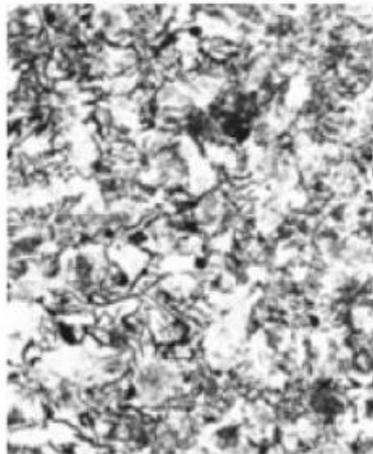
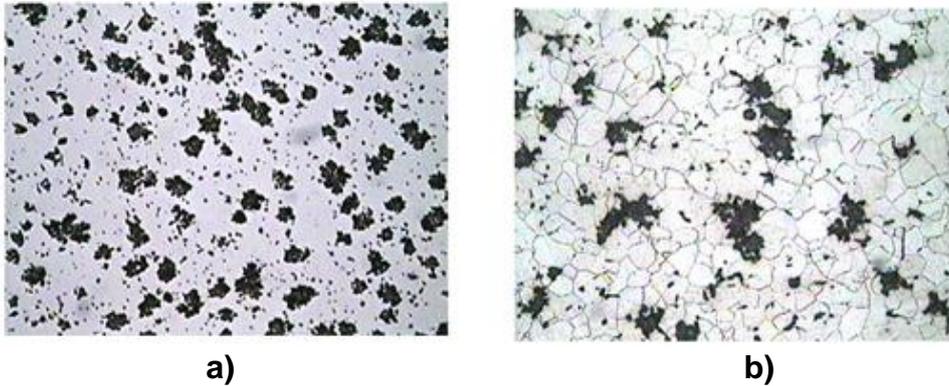


Figura 28: Ferro fundido mesclado
Fonte: Adaptada de (DE PEIXOTO et al., 2015)

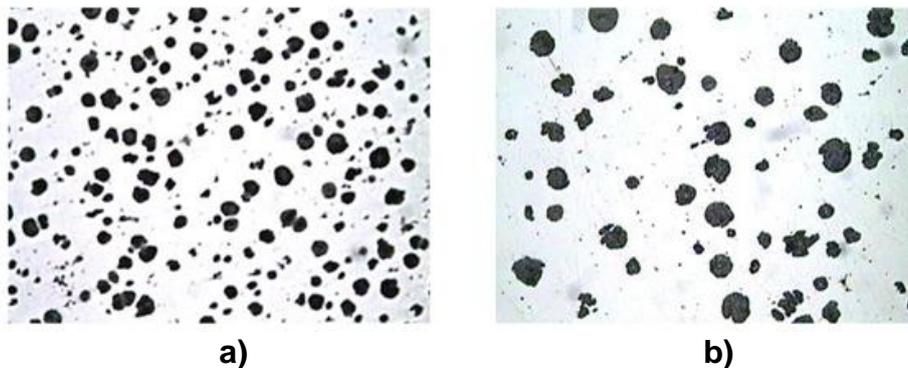
Ferro fundido maleável: Oriundo do ferro fundido branco, através do tratamento térmico de maleabilização, onde o objetivo desse tratamento é transformar a cementita em grafita esferoidal, proporcionando melhores propriedades mecânicas como maior resistência, limite de escoamento e

ductilidade. Sua coloração na superfície da fratura é apresentada como cinza claro (VIDAL, 2013) e (COSTA, 2003).



a) **b)**
Figura 29: Ferro fundido maleável - a) Microestrutura com zoom de 100x
b) Microestrutura com zoom de 200x
Fonte: Adaptada de (SOUZA, 2019)

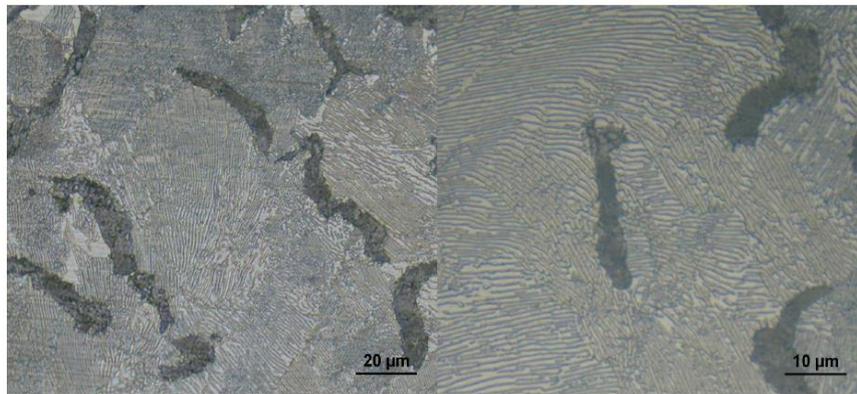
Ferro fundido nodular: É caracterizado pela presença do carbono livre na forma de nódulos grafíticos, proporcionando a esse tipo de ferro fundido características mecânicas superiores à do ferro fundido maleável. Sua aplicação pode ser observada em eixos de comando de válvulas, carcaças, peças de suspensão, tubulações entre outros. A sua coloração na superfície de fratura é apresentada como prateada (COSTA, 2003) e (VIDAL, 2013).



a) **b)**
Figura 30: Ferro fundido nodular - a) Microestrutura com zoom de 100x
b) Microestrutura com zoom de 200x
Fonte: Adaptada de (SOUZA, 2019)

Ferro fundido vermicular: Conhecido também como ferro fundido de grafita compacta, o ferro fundido vermicular apresenta carbono livre na forma de grafita. É obtido através da adição do titânio com a finalidade de abaixar o nódulo de grafita. Suas propriedades estão entre os ferros fundidos nodular e cinzento, possuindo uma aplicação ampla como coletores de exaustão, blocos de motores a diesel de novas gerações, caixas de engrenagens entre outros.

Porém ainda é novo no mercado. Sua coloração na superfície de fratura é apresentada como mesclada (COSTA, 2003) e (VIDAL, 2013).



a)

b)

**Figura 31: Ferro fundido vermicular - a) Microestrutura com zoom de 500x
b) Microestrutura com zoom de 10000x**

Fonte: Adaptada de (MASSIRER JUNIOR; GUESSER, 2011)

3. MATERIAIS E MÉTODOS

A principal ferramenta do presente trabalho foi desenvolvida através do software Microsoft Visual Studio utilizando a edição para estudantes conhecida como Visual Community, escrevendo o código com a linguagem de POO (Programação Orientada a Objetos) C++. Devido à alta compatibilidade com vários sistemas e também a possibilidade de incorporar e/ou alterar novas funcionalidades (FELICIANO; LAMEGO, 2008), optou-se por desenvolver o código com essa linguagem de programação. Outro fator importante para o emprego dessa linguagem é que esta possui uma ampla literatura e vários trabalhos publicados que nos auxiliam no desenvolvimento da ferramenta computacional. Algumas funções da biblioteca do software OpenCV foram utilizadas devido ao seu poder de processamento de imagens e vídeos (OPENCV, 2021). O objetivo principal do trabalho é aplicar e validar um código computacional serial com finalidade de gerar imagens em 3D de forma fidedigna por meio de imagens em 2D extraídas de um processo de seccionamento em série, seja ele destrutivo ou não.

A visualização da reconstrução 3D foi através das ferramentas Tecplot e ParaView e a validação pelo ImageJ. Sendo assim, de forma sintetizada, observa-se na figura 32 um diagrama esquemático da metodologia básica que foi empregada no presente trabalho. Ressalta-se que processo de análise das imagens é o mesmo, sendo que, o que de fato diferencia é a forma de aquisição dessas imagens, através de seccionamento em série destrutivo e não destrutivo.

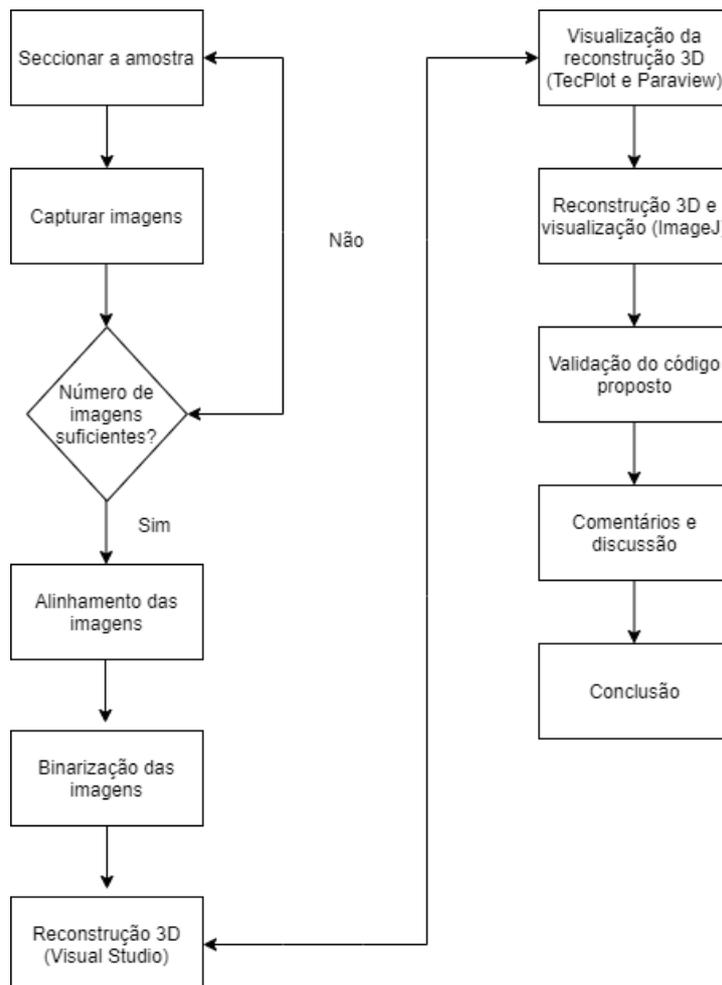


Figura 32 : Passo a passo da metodologia empregada
Fonte: Autoria própria (2021)

De forma detalhada, observa-se o passo a passo da metodologia mostrada na figura 32:

As amostras utilizadas passaram por um processo de seccionamento em série no qual foi seccionada e a cada seção foi capturada uma imagem até a leitura completa da amostra de ferro fundido (seccionamento em série destrutivo) e/ou paralelepípedo de alumínio envolvido em resina (seccionamento em série não destrutivo) para o desenvolvimento da simulação computacional. Salienta-se que esses processos foram realizados, respectivamente, pelo equipamento Robo-Met para imagens de ferro fundido e V-tomex-m para imagens do paralelepípedo de alumínio envolvido em resina.

O próximo passo consiste em alinhar as imagens para evitar possíveis erros durante a renderização e depois binarizá-las para que o computador possa interpretar os dados descritos possibilitando a identificação dos objetos

e/ou materiais separando-os do fundo da imagem. Os processos de alinhamento e binarização foram realizados pelos equipamentos Robo-Met para o ferro fundido e V-tomex-m para o paralelepípedo de alumínio envolvido em resina.

Finalizando a binarização, a reconstrução 3D pode ser iniciada através do Microsoft Visual Studio e posteriormente visualizada com o auxílio dos programas Tecplot e ParaView.

Após a simulação computacional, foi possível identificar de forma tridimensional as características inerentes das amostras de estudos.

Então, o próximo passo é validar o código utilizando o ImageJ, pois este software é amplamente utilizado na literatura e já possui a sua respectiva confiabilidade.

E por último, os comentários e discussões foram realizados para informar que o algoritmo proposto funciona conforme esperado.

3.1 Equipamentos utilizados

No presente trabalho, utilizou-se dois diferentes tipos de materiais para dois tipos de técnica de seccionamento em série, sendo o ferro fundido nodular para o seccionamento em série de forma destrutiva e o outro material um alumínio, com formato de paralelepípedo envolvido com resina em forma de tarugo para o seccionamento em série de forma não destrutiva. Assim, detalha-se o funcionamento de cada técnica abordada.

3.1.1 Seccionamento em série de forma destrutiva

Ressalta-se que as imagens oriundas da técnica de seccionamento em série de forma destrutiva foram disponibilizadas pelo Ph.D. Veeraraghavan Sundar.

Dessa forma, para o seccionamento em série da amostra de ferro fundido nodular, foi utilizado o Robot-Met que é um sistema de corte em série totalmente automatizado para investigações microestruturais tridimensionais. Inventado em parceria com o laboratório de pesquisa da força aérea dos Estados Unidos, o funcionamento do Robo-Met consiste em desbastar e alinhar automaticamente e de forma sequencial as camadas do material com precisão

a nível de microns, permitindo a visualização metalográfica e a geração de imagens em 2D das microestruturas dos materiais. Após a etapa de desbaste, o pós-processamento do equipamento faz a reconstrução dessas imagens 2D, reunindo-as em uma escala óptica, em modelos 3D. O equipamento fornece uma interface fácil de usar, sendo que o software de automação do processo de seccionamento vem pré-carregado e pronto para uso, permitindo uma configuração amigável.

O Robo-Met é alojado em um laboratório totalmente fechado, livre de detritos e projetado para segurança. Uma superfície de trabalho em granito fornece um suporte estrutural e uma plataforma para as imagens sem vibração. Essa tecnologia de ponta oferece a mais alta qualidade e confiabilidade, permitindo a análise da porosidade, distribuição dos grãos, tamanhos dos grãos, orientação das fibras, propriedades da estrutura e a investigação de fraturas nos pontos de estresse são trazidas com maior nitidez.

Esse equipamento comparado ao processamento tradicional de materiais, pode aumentar as taxas de corte em até 100 vezes, fornecendo precisão e velocidade em paralelo, sendo que o Robo-Met é totalmente automatizado e programável com altas precisões. Salieta-se que o equipamento traz economia de mão de obra e tempo para os processamentos de materiais, além de consistência e confiança nos dados fornecidos (UES, 2021).



Figura 33: Imagem do Robo-Met
Fonte: (UES, 2021)

3.1.2 Seccionamento em série de forma não destrutiva

Ressalta-se que as imagens oriundas da técnica de seccionamento em série de forma não destrutiva foram disponibilizadas pelo M.Sc. Wyslân Jefferson Lima Garção.

Assim, para o seccionamento em série do alumínio (comercialmente puro - AA1050) com formato de paralelepípedo envolvido em material de resina em forma de tarugo, foi utilizado o equipamento de microCT, cujo modelo é V-tomex-m da General Electric (GE) para a aquisição das imagens em 2D, sendo que este é um sistema totalmente automatizado para investigações microestruturais utilizando o sistema de microtomografia computadorizada para obtenção das imagens bidimensionais (TELES, 2016).

O microtomógrafo possui tubos de raios X, com ajustes de tensão e corrente variáveis em função do material de estudo, sendo que a amostra é rotacionada para que seja possível analisar o material em 360°, obtendo assim informações em três dimensões, possibilitando análises mais detalhadas.

A máquina possui dois tubos de raios X, sendo um de nanofoco com tensão de até 180kV e potência de até 15W e outro de microfoco com tensão de até 300kV e potência até 500W, permitindo análises de materiais com diversos tamanhos e pesos (MOREIRA, 2021). Para maiores detalhes do equipamento, ver referências (TELES, 2016) e (MOREIRA, 2021).

Para a obtenção das imagens, o sistema possui um detector capaz de converter os feixes de raios X em imagens digitais em duas dimensões (TELES, 2016).

Observa-se na figura 34 o microtomógrafo utilizado para aquisição das imagens.



Figura 34: Imagem do Phoenix V-tome|x-m
Fonte: (MOREIRA, 2021)

3.2 Estrutura simplificada do programa

O programa proposto, responsável por fazer a reconstrução 3D, está estruturado em classes nos quais estão distribuídas na seguinte ordem: Dados, Imagem, Matriz, Principal e Saída, sendo todas utilizando .cpp como extensão.

3.2.1 Classe de Dados

Esta parte do programa é responsável por fazer a coleta e armazenamento dos dados que são definidos de uma maneira inicial em um arquivo de entrada, seguindo um padrão estruturado intitulado como Entrada.dat, permitindo assim o preenchimento dos dados de forma padronizada.

3.2.2 Dados de entrada

Para fazer a reconstrução tridimensional, tomando como base fotografias em 2D, é necessário que o algoritmo receba dados iniciais a fim de possibilitar a simulação, são eles:

Tipo de arquivo de saída: Nessa aba é possível escolher o tipo de arquivo de saída (.dat) que será utilizado para leitura pelos programas de visualização gráfica, tais como: Tecplot e ParaView.

Nome do arquivo de saída: É a parte do algoritmo responsável pelo nome do arquivo de saída escolhido pelo usuário.

Extensão do arquivo imagem: Local onde é possível informar ao algoritmo a extensão das imagens que serão utilizadas na simulação, tais como: .TIF, .JPEG, .PNG entre outros.

Diretório que contém as imagens: É o caminho no qual se encontra a pasta com as imagens a serem utilizadas.

Tipo de método a ser executado, média das cores: Essa aba é responsável pela definição do método de preenchimento dos vazios entre as imagens, média das cores, no qual foi definida no trabalho.

Número inicial de imagens: É a parte do programa onde se informa a quantidade de imagens bidimensionais que serão utilizadas para fazer a reconstrução 3D.

3.2.3 Classe imagem

Esta classe é responsável por reconhecer as imagens a serem utilizadas e conseqüentemente possibilitar a leitura de suas propriedades, tais como: quantidade de pixels e intensidade de cores. Para possibilitar e facilitar a leitura e interpretação das imagens, utilizou-se o programa OpenCV devido a sua capacidade de carregar tais imagens e a sua facilidade em trabalhar com o Visual Studio além da sua capacidade de processamento propiciando um carregamento das imagens com maior velocidade.

3.2.4 Classe matriz

O NMM (Núcleo de Modelamento Microestrutural) desenvolveu a classe matriz na qual possibilitou a implementação de uma malha tridimensional cúbica, criando assim um domínio discreto. Segundo (GRAÇA, 2019), a célula que possui um formato similar ao cubo é a menor unidade dessa malha, cuja união de forma ordenada gera uma matriz, representando o espaço discreto. Esta classe desenvolvida é responsável por criar um vetor unidimensional como estrutura de dados.

3.2.5 Classe principal

O código desenvolvido faz a simulação utilizando, para o preenchimento dos vazios, a técnica de média das cores. Neste método são planos intermediários que possibilitam o preenchimento dos vazios existentes entre as imagens 2D após o seccionamento em série através da média das cores, que pode ser obtida através da soma dos valores de R, G e B e depois dividir cada um por dois. Essa técnica consiste basicamente em empilhar as imagens bidimensionais, no qual o algoritmo reconhece como uma matriz de duas dimensões, formando assim, após o empilhamento das imagens, uma matriz tridimensional que, de fato, é a reconstrução 3D (GRAÇA, 2019).

Segundo (GRAÇA, 2019), para iniciar o processo de média das cores é necessário informar a quantidade de imagens e o número de planos a ser analisados. Nessa etapa do algoritmo, ele trabalha em conjunto com o OpenCV utilizando a função `inread()` do Visual Studio. A partir dessas informações, é possível calcular o vetor intensidade, a quantidade de pontos, número de camadas entre outros.

3.2.6 Classe saída

Esta classe é responsável pela geração de um arquivo de texto com extensão `.dat` que possibilita a identificação dos pontos espaciais com as suas intensidades e cores definidas pelo sistema RGB. Com esse documento de texto de saída do programa `.dat` é possível visualizar as imagens reconstruídas após a simulação através de ferramentas gráficas, tais como o Tecplot, ParaView ou programas similares que possibilitam a leitura de arquivos `.dat`.

Para validar o código computacional, foi realizada uma nova simulação utilizando o software ImageJ, pois o mesmo possui diversas aplicações em trabalhos que podem ser encontrados na literatura e com bastante confiabilidade. Após a realização de todas as etapas propostas, foi possível comentar e discutir os resultados obtidos através da simulação computacional.

3.3 Informações do computador

O computador utilizado para a pesquisa, desenvolvimento da proposta/dissertação, artigos, bem como, desenvolvimento do algoritmo,

geração dos arquivos de saída, obtenção da visualização através de ferramentas gráficas, realização da reconstrução 3D e análise dos resultados obtidos foi um computador pessoal (notebook) com características conforme quadro 1.

Informações do Computador	
Marca	Dell
Modelo	Vostro
Sistema Operacional	Windows 10 Home Single Language
Processador	Intel® Core™ i5
Memória RAM	4 GB
Placa de Vídeo	Integrada
Velocidade	2,60GHz

Quadro 1: Informações do computador pessoal
Fonte: Autoria própria (2021)

4. RESULTADOS E DISCUSSÃO

Com a aplicação do código proposto, utilizando as imagens de ferro fundido nodular, oriundas do seccionamento em série de forma destrutiva e também da aplicação do código nas imagens de alumínio com formato de paralelepípedo envolvido em material de resina com forma de tarugo, oriundas do seccionamento em série de forma não destrutiva, obteve-se como resultado os volumes reconstruídos de acordo com cada tipo de material no presente trabalho.

Verificou-se, através das inserções das informações apropriadas para a entrada de dados, que foi possível estabelecer uma comparação entre o código desenvolvido no Visual Studio e também no ImageJ, possibilitando a validação do algoritmo independente da aplicação, seja ela utilizando o seccionamento em série de forma destrutiva ou não destrutiva.

4.1 Reconstrução 3D utilizando as imagens do Robo-Met

Inicialmente foi necessário inserir os dados necessários na aba de entrada do Visual Studio para que o algoritmo conseguisse fazer a reconstrução tridimensional corretamente. As informações indicadas na figura 35 são essenciais, uma vez que, sem elas o algoritmo não funciona corretamente.

4.1.1 Aplicação do algoritmo proposto nas imagens de ferro fundido

Como dito anteriormente, o algoritmo precisa de algumas informações de entrada que são: tipo de arquivo de saída, nome do arquivo de saída, extensão das imagens, diretório e o número de imagens a serem analisadas. Para o presente trabalho, foram utilizadas 200 imagens de ferro fundido nodular com extensão em tif, a técnica utilizada para o preenchimento dos vazios entre as imagens foi a média das cores e o número de planos entre as imagens foi igual a 1. Esses dados de entradas podem ser observados na figura 35.

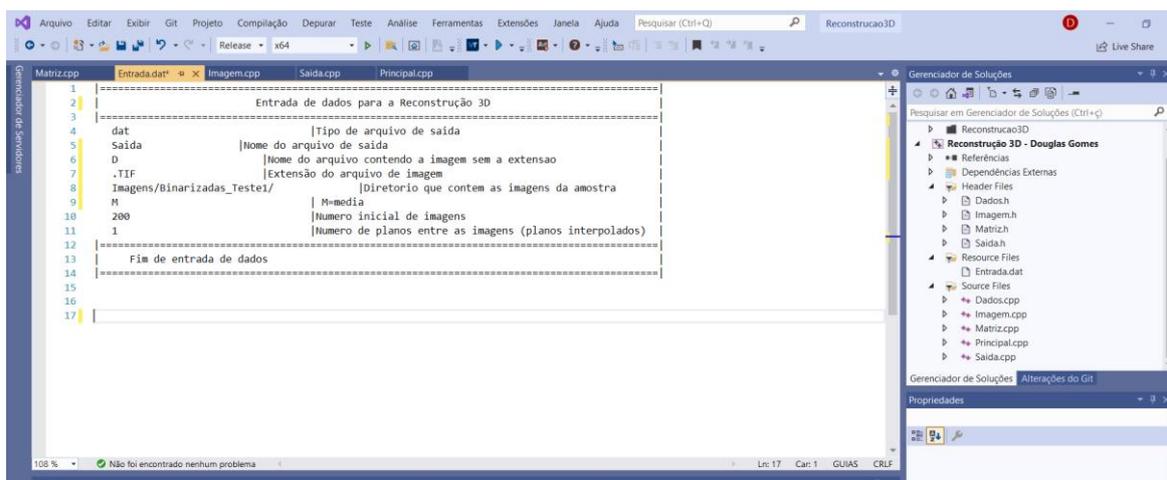


Figura 35: Tela inicial de inserção de dados
Fonte: Autoria própria (2021)

De posse dos dados iniciais, começa-se o segundo passo que é a depuração do código. Após esse passo, o programa imprime na tela algumas informações através do prompt de comando do Windows, que nesse caso é o console de depuração do Visual Studio. Pode-se verificar esse segundo passo na figura 36.

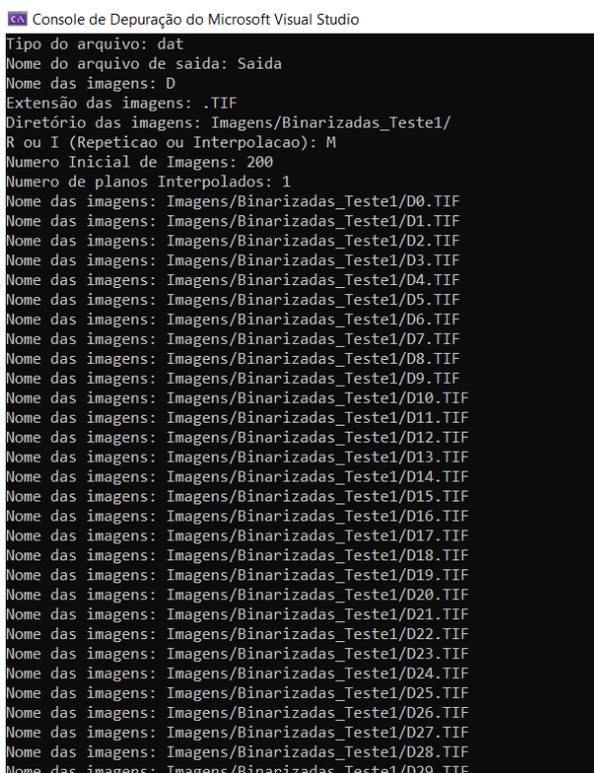


Figura 36: Console 1 de depuração Visual Studio
Fonte: Autoria própria (2021)

Um detalhe muito importante é que o algoritmo faz uma análise dos pixels das imagens, ou seja, linhas e colunas. Dessa forma, para que seja possível empilhar as imagens bidimensionais, que nesse caso são matrizes, e transformá-las em matrizes tridimensionais é necessário percorrer pixel a pixel de todas as imagens bidimensionais, sendo assim, na figura 36, 37 e 38 é mostrado o carregamento de imagem a imagem até a quantidade total de arquivos informadas na entrada de dados.

```
Console de Depuração do Microsoft Visual Studio
Nome das imagens: Imagens/Binarizadas_Testes1/D184.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D185.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D186.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D187.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D188.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D189.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D190.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D191.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D192.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D193.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D194.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D195.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D196.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D197.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D198.TIF
Nome das imagens: Imagens/Binarizadas_Testes1/D199.TIF
Número de Linhas: 239
Número de Colunas: 234
Número de Planos: 200
Tamanho do vetor intensidade: 11185200
Quant Total de Planos: 399
contador: 111852
contador: 223704
contador: 335556
contador: 447408
contador: 559260
contador: 671112
contador: 782964
contador: 894816
contador: 1006668
contador: 1118520
contador: 1230372
contador: 1342224
contador: 1454076
contador: 1565928
contador: 1677780
contador: 1789632
contador: 1901484
```

Figura 37: Console 2 de depuração Visual Studio
Fonte: Autoria própria (2021)

Após o carregamento de todas as imagens e finalização da reconstrução 3D, o console de depuração imprime na tela o local do arquivo de saída, ou seja, o seu respectivo diretório. A figura 38 mostra o diretório do arquivo no final da depuração.

```
Console de Depuração do Microsoft Visual Studio
novo_z:184
novo_z+rounded_down:184.5
novo_z:185
novo_z+rounded_down:185.5
novo_z:186
novo_z+rounded_down:186.5
novo_z:187
novo_z+rounded_down:187.5
novo_z:188
novo_z+rounded_down:188.5
novo_z:189
novo_z+rounded_down:189.5
novo_z:190
novo_z+rounded_down:190.5
novo_z:191
novo_z+rounded_down:191.5
novo_z:192
novo_z+rounded_down:192.5
novo_z:193
novo_z+rounded_down:193.5
novo_z:194
novo_z+rounded_down:194.5
novo_z:195
novo_z+rounded_down:195.5
novo_z:196
novo_z+rounded_down:196.5
novo_z:197
novo_z+rounded_down:197.5
novo_z:198
novo_z+rounded_down:198.5
novo_z:199
novo_z+rounded_down:199.5

0 F:\Mestrado UFF dissertação 2021\Reconstrucao 3D-Versão final Testes\x64\Release\Reconstrução 3D - Douglas Gomes.exe (processo 12048) foi encerrado com o código 0.
Pressione qualquer tecla para fechar esta janela...
```

Figura 38: Console 3 de depuração Visual Studio
Fonte: Autoria própria (2021)

Após esses passos, o Visual Studio gera um arquivo de saída com o nome que pode ser editado a gosto do usuário. Sendo assim, após esse procedimento é possível visualizar o arquivo .dat no próprio diretório de arquivo. Esse arquivo cuja extensão é .dat foi utilizado pelos softwares de visualização gráfica, Tecplot e ParaView, para imprimir na tela o volume gerado pela reconstrução tridimensional. Dessa forma, na figura 39, observa-se o volume impresso na tela com a utilização do Tecplot.

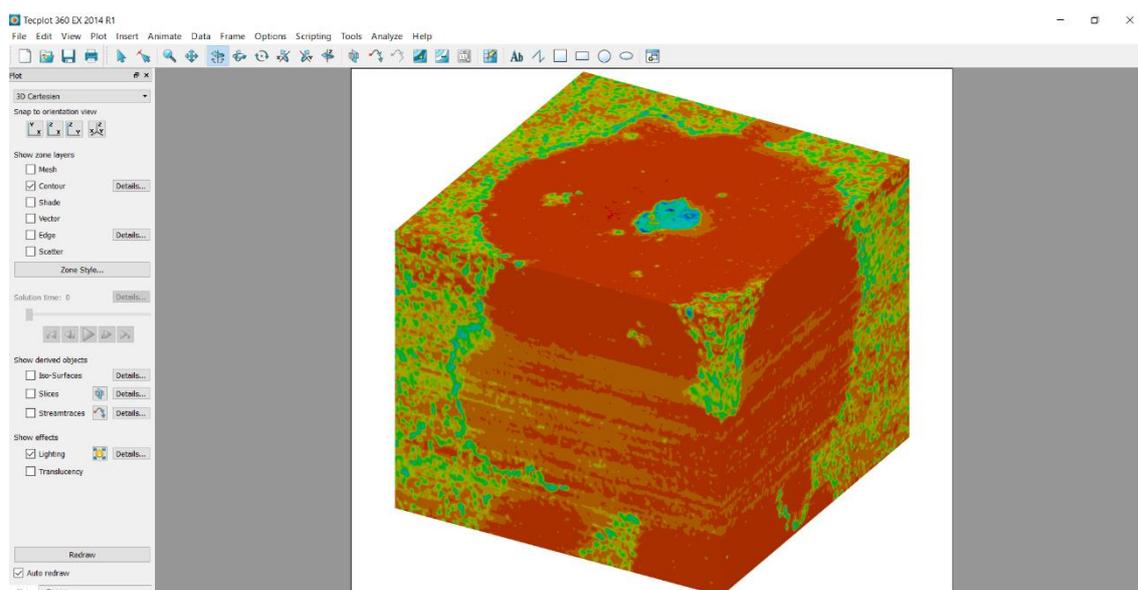


Figura 39: Reconstrução tridimensional - Tecplot
Fonte: Autoria própria (2021)

E agora, para efeitos de comparação, observa-se na figura 40, o volume impresso na tela com a utilização do ParaView.

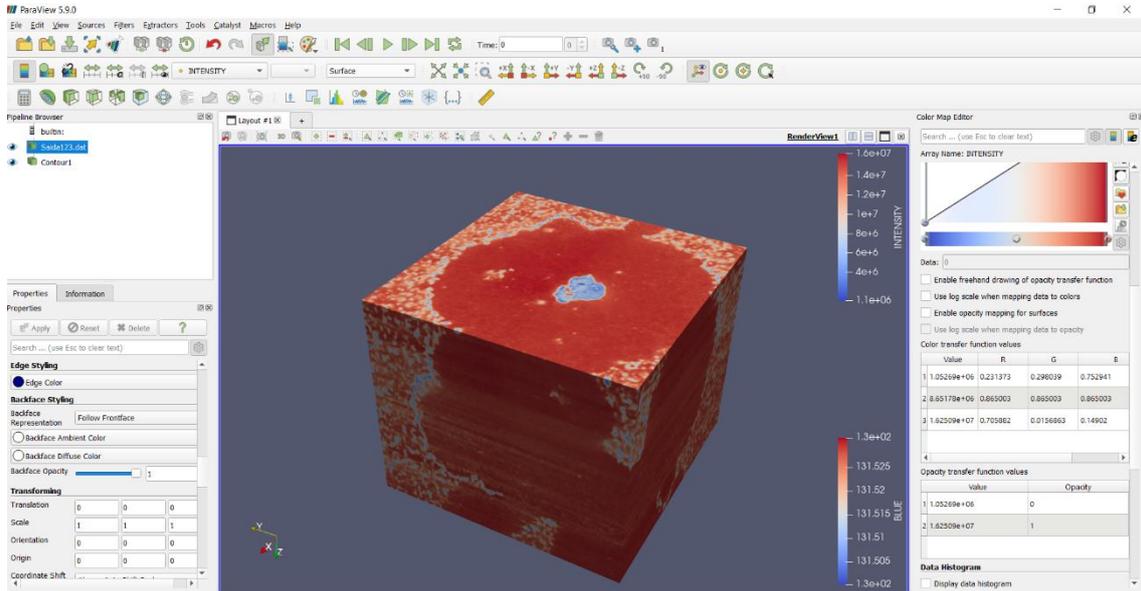


Figura 40: Reconstrução tridimensional - ParaView
Fonte: Autoria própria (2021)

4.1.2 Validação do algoritmo proposto – ferro fundido

Para comprovar que o algoritmo funciona corretamente, utilizou-se o ImageJ, pois este possui uma vasta literatura que nos auxilia com a comprovação do algoritmo desenvolvido. Observe na figura 41 a imagem gerada pelo programa proposto para validação após o carregamento imagens.

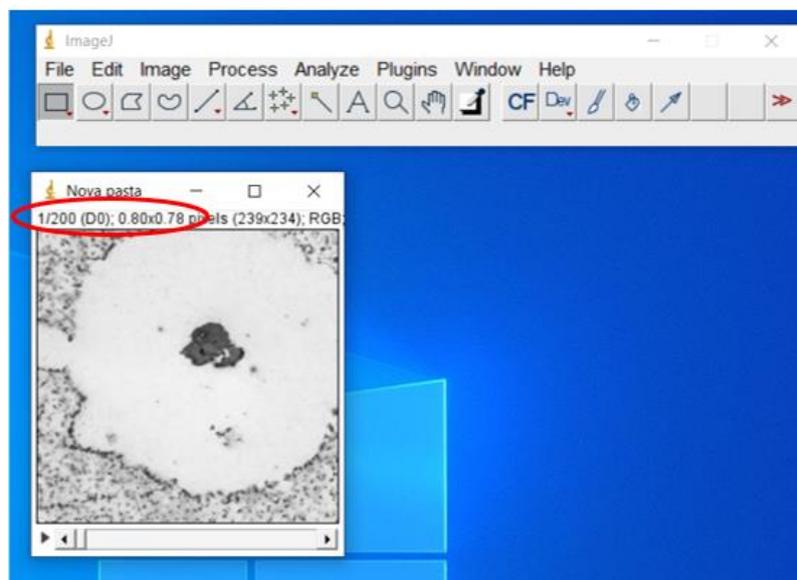


Figura 41: Carregamento das 200 imagens
Fonte: Autoria própria (2021)

Diferentemente do Tecplot e ParaView, primeiramente o ImageJ carrega todas as imagens solicitadas pelo usuário e imprime na tela. Dessa forma, para fazer a reconstrução tridimensional é necessário realizar mais um passo. Verifica-se na figura 42 as 200 imagens utilizadas no trabalho proposto carregadas pelo programa e também o passo necessário para fazer a reconstrução 3D.

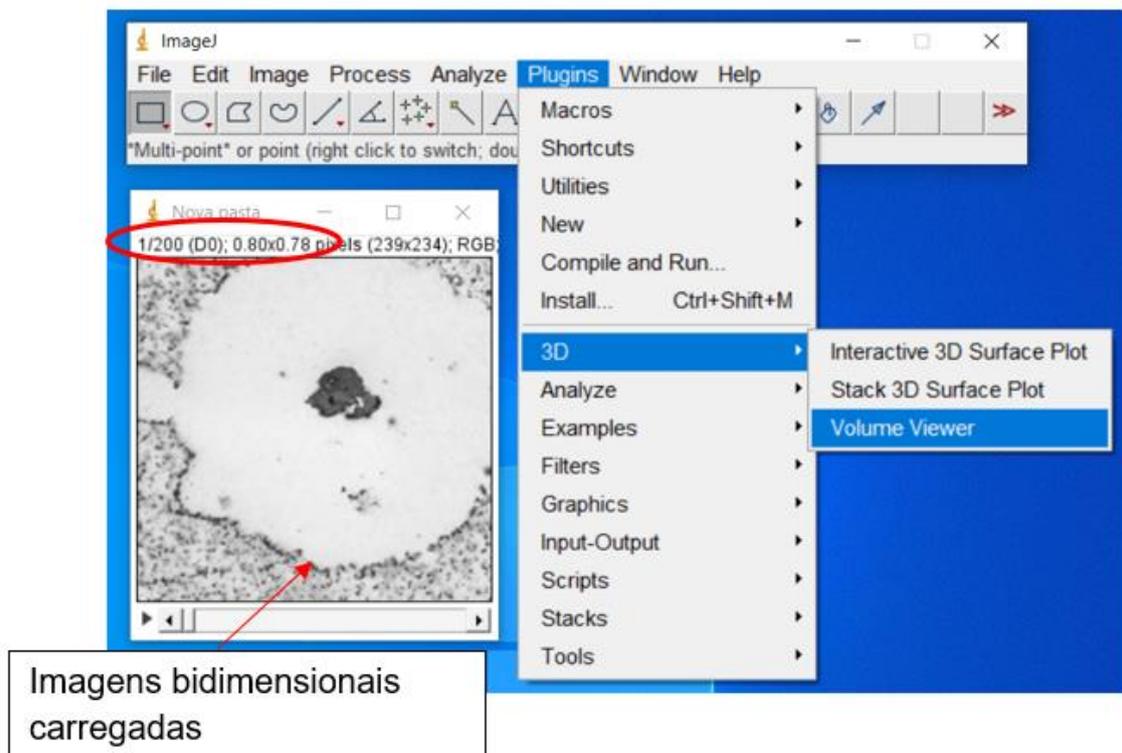


Figura 42: Geração do volume – ferro fundido
Fonte: Autoria própria (2021)

Então, para comprovar que o algoritmo é eficaz e confiável, nota-se na figura 43, que a imagem tridimensional gerada é similar as imagens geradas pelos dois programas de visualização gráfica, Tecplot e ParaView. Salienta-se que esses programas de visualização imprimem na tela volume cuja a extensão é oriunda do arquivo .dat gerado pelo Visual Studio. Ressalta-se que tanto o TecPlot quanto o ParaView podem abrir arquivos com outras extensões, porém para o presente trabalho, foi utilizada a extensão .dat, oriunda do Visual Studio. Sendo assim, observa-se na figura 43 o volume gerado pelo ImageJ.

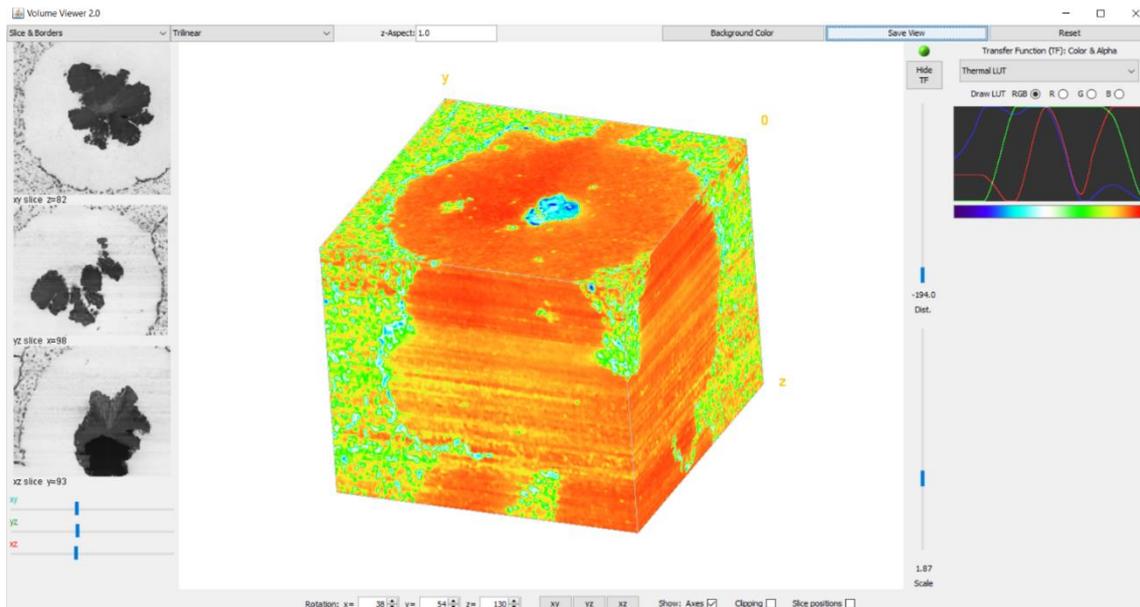


Figura 43: Volume gerado pelo ImageJ – ferro fundido
Fonte: Autoria própria (2021)

Dessa forma, para uma melhor visualização da reconstrução tridimensional do ferro fundido nodular, observa-se na figura 44 os volumes apresentados em cada software.

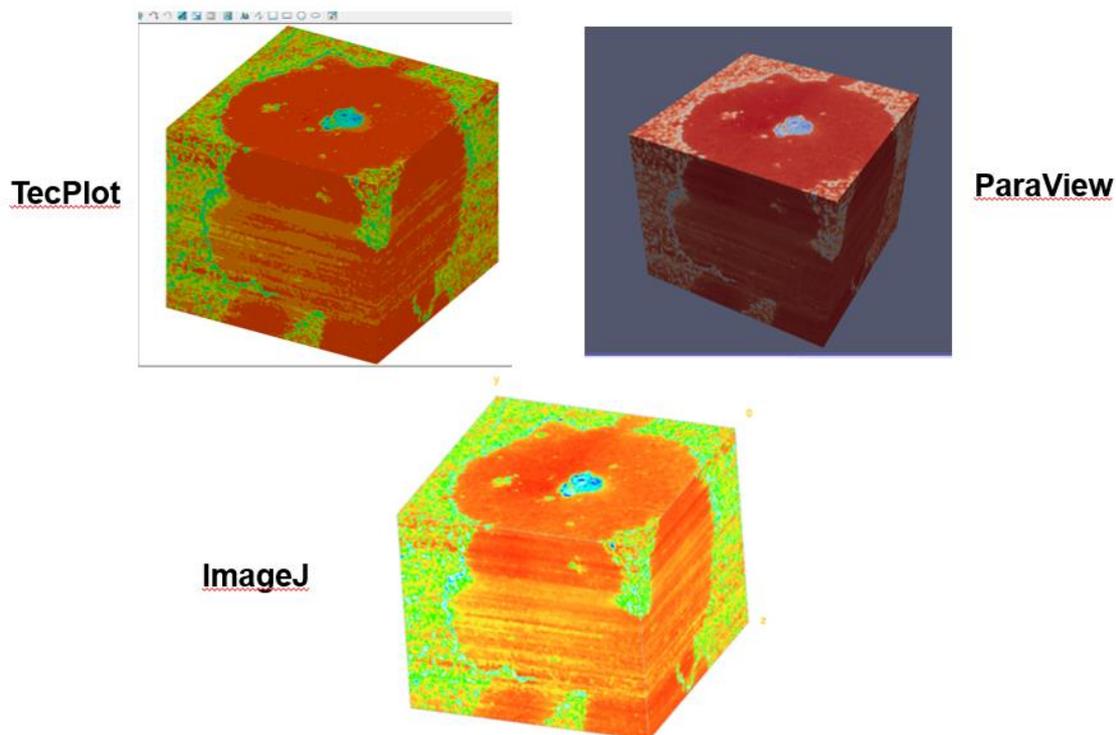


Figura 44: Ferro fundido – TecPlot, Paraview e ImageJ
Fonte: Autoria própria (2021)

Outra forma de comparar a reconstrução 3D do ferro fundido nodular, além do ImageJ, foi o nódulo reconstruído pelo RoboMet, imagem essa disponibilizada pelo Sundar Veeraraghavan, no qual é possível observar todos as imagens em 3D visualizadas através dos programas de interface gráfica. Assim, segue abaixo na figura 45, a visualização dos nódulos de ferro fundido nodular.

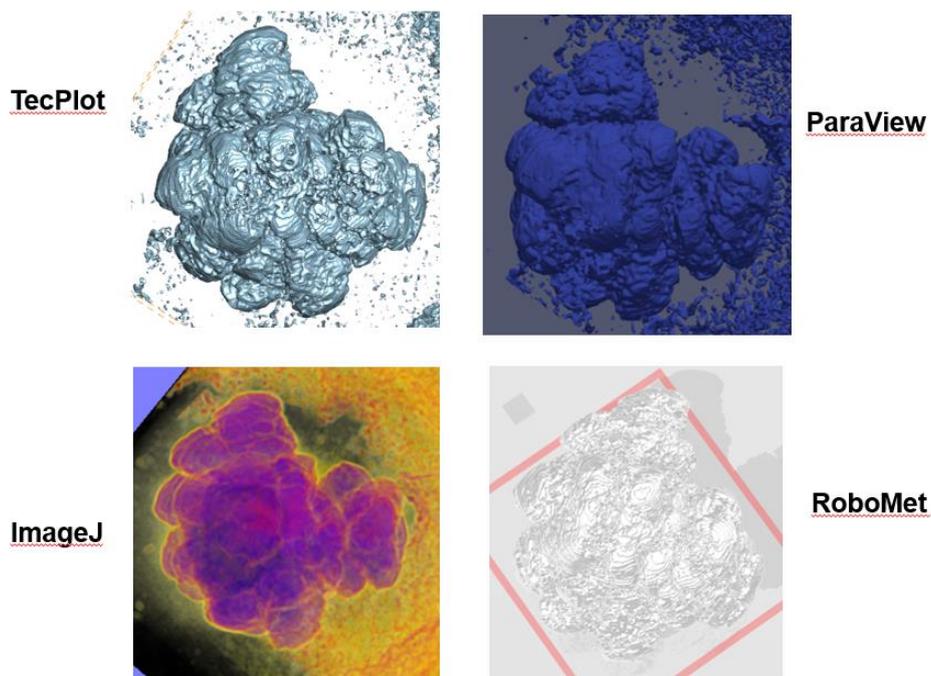


Figura 45: Nódulo de ferro fundido – TecPlot, Paraview e ImageJ
Fonte: Autoria própria (2021)

Salienta-se que as imagens de ferro fundido utilizadas no trabalho têm uma matriz de 239 x 234 pixels e que foi necessário diminuir o tamanho do arquivo de imagem por conta da limitação do computador, pois quanto maior a resolução maior será o arquivo de trabalho e conseqüentemente é exigido um elevado poder de processamento. As imagens iniciais, oriundas do Robo-Met, equipamento responsável pelo seccionamento em série de forma destrutiva do ferro fundido, possuem em cada imagem uma matriz de 798 x 780 pixels.

4.2 Reconstrução 3D utilizando as imagens do microtomógrafo V-tomex-m

Como mostrado no capítulo 4.1, para que o algoritmo proposto funcione corretamente, é necessário inserir todos os dados necessários na aba de

entrada do Visual Studio para que o código faça a reconstrução 3D conforme esperado.

4.2.1 Aplicação do algoritmo proposto nas imagens do paralelepípedo de alumínio envolvido em resina

Assim como na análise realizada na aplicação do algoritmo nas imagens de ferro fundido, o código proposto necessita de informações para garantir o seu correto funcionamento, que no caso são: tipo de arquivo de saída, nome do arquivo de saída, extensão das imagens, diretório e o número de imagens a serem analisadas. O número de imagens utilizadas para análise do paralelepípedo de alumínio envolvido em resina foi de 74 com extensão em jpg, a técnica utilizada para o preenchimento dos vazios foi a da média das cores e o número de planos entre as imagens foi igual a 1. Salienta-se que, por conta da limitação de processamento do computador utilizado no presente trabalho e também pelo tamanho de cada imagem, foi necessário reduzir a quantidade de pixels dos arquivos de forma que não comprometesse a qualidade da reconstrução tridimensional. Diante do exposto, segue na figura 46 os dados utilizados no Visual Studio.

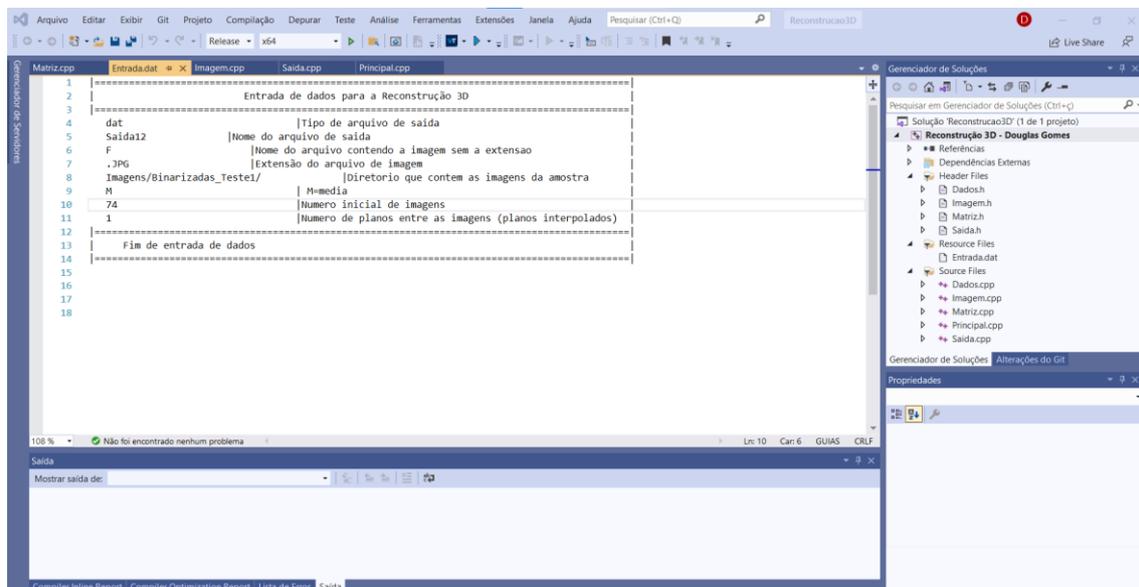
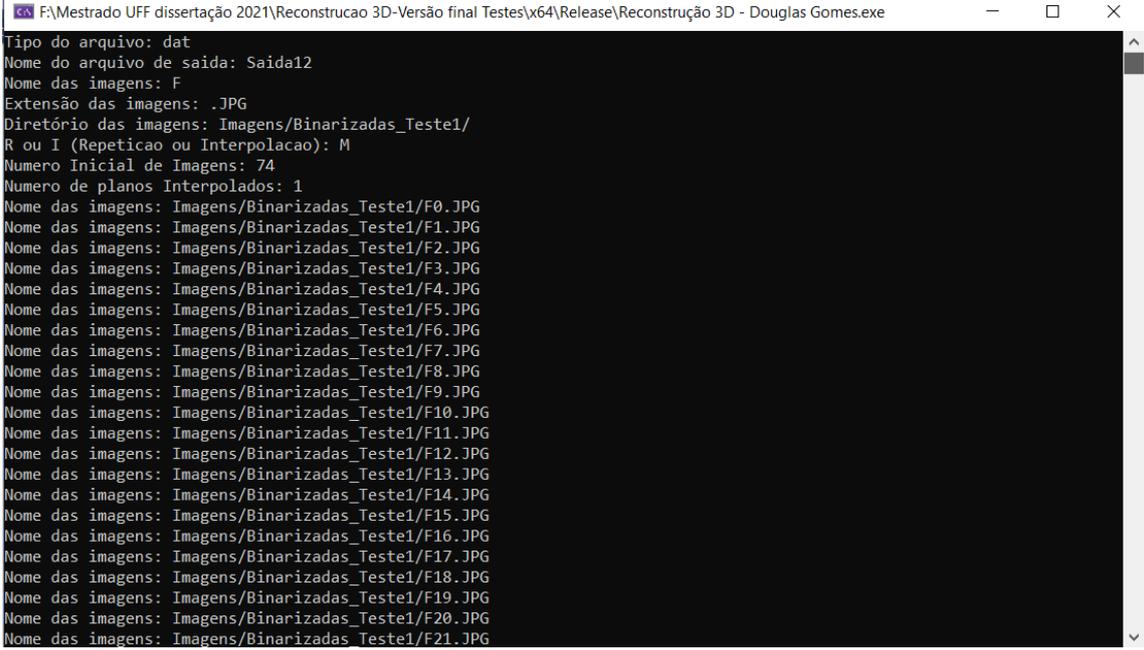


Figura 46: Aba de inserção de dados
Fonte: Autoria própria (2021)

A partir do momento que o código obtém os dados iniciais, inicia-se o próximo passo que é a depuração do algoritmo. Após essa etapa, o programa

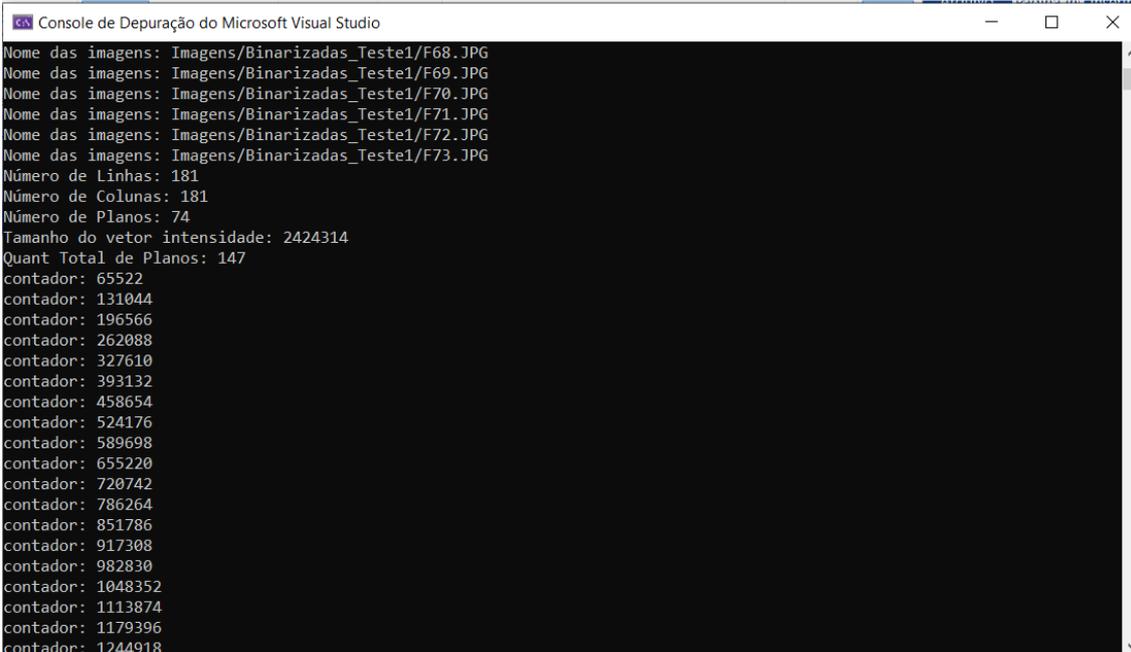
imprime na tela as informações a partir do prompt de comando do Windows. Verifica-se essa etapa na figura 47.



```
F:\Mestrado UFF dissertação 2021\Reconstrucao 3D-Versão final Testes\x64\Release\Reconstrução 3D - Douglas Gomes.exe
Tipo do arquivo: dat
Nome do arquivo de saída: Saida12
Nome das imagens: F
Extensão das imagens: .JPG
Diretório das imagens: Imagens/Binarizadas_Testes1/
R ou I (Repeticao ou Interpolacao): M
Numero Inicial de Imagens: 74
Numero de planos Interpolados: 1
Nome das imagens: Imagens/Binarizadas_Testes1/F0.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F1.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F2.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F3.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F4.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F5.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F6.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F7.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F8.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F9.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F10.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F11.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F12.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F13.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F14.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F15.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F16.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F17.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F18.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F19.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F20.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F21.JPG
```

Figura 47: Console 4 de depuração Visual Studio
Fonte: Autoria própria (2021)

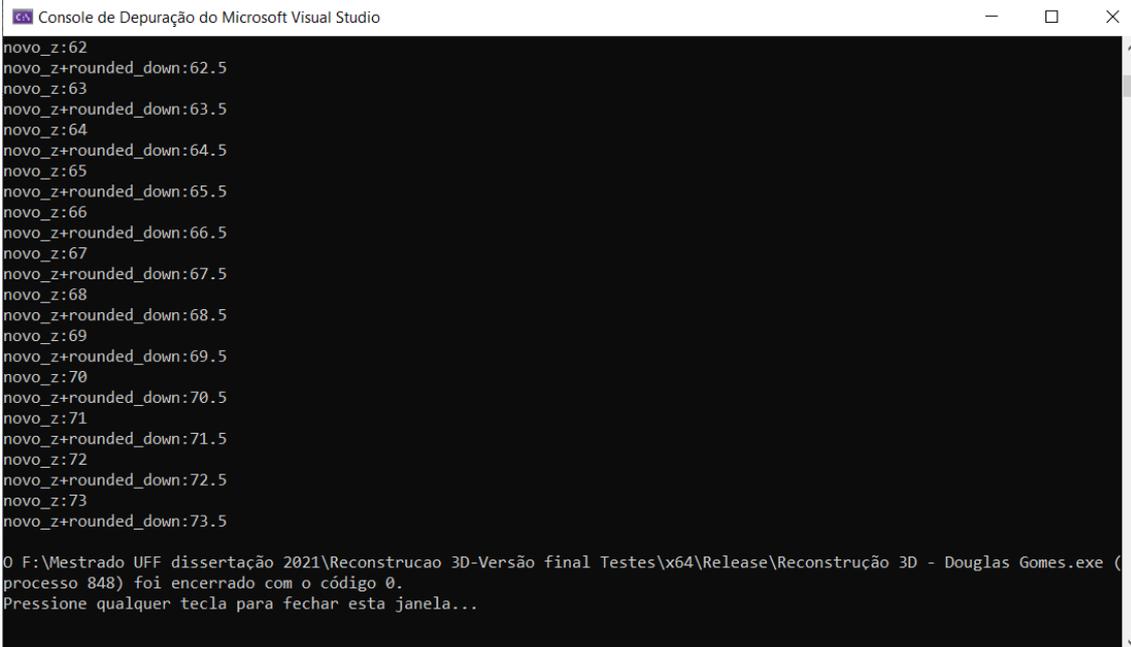
Ressalta-se que o código faz uma análise dos pixels de todas as imagens, ou seja, linhas e colunas. Assim, para que seja possível a realização da reconstrução 3D, é necessário que o código percorra pixel a pixel para garantir que as imagens estão sendo analisadas corretamente. O carregamento total das imagens pode ser observado nas figuras 47, 48 e 49.



```
Console de Depuração do Microsoft Visual Studio
Nome das imagens: Imagens/Binarizadas_Testes1/F68.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F69.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F70.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F71.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F72.JPG
Nome das imagens: Imagens/Binarizadas_Testes1/F73.JPG
Número de Linhas: 181
Número de Colunas: 27
Número de Planos: 74
Tamanho do vetor intensidade: 2424314
Quant Total de Planos: 147
contador: 65522
contador: 131044
contador: 196566
contador: 262088
contador: 327610
contador: 393132
contador: 458654
contador: 524176
contador: 589698
contador: 655220
contador: 720742
contador: 786264
contador: 851786
contador: 917308
contador: 982830
contador: 1048352
contador: 1113874
contador: 1179396
contador: 1244918
```

Figura 48: Console 5 de depuração Visual Studio
Fonte: Autoria própria (2021)

Após carregar todas as imagens, o console de depuração imprime na tela o diretório do arquivo que será utilizado nas ferramentas de visualização gráficas. É possível ver na figura 49 o local do arquivo no computador.



```
Console de Depuração do Microsoft Visual Studio
novo_z:62
novo_z+rounded_down:62.5
novo_z:63
novo_z+rounded_down:63.5
novo_z:64
novo_z+rounded_down:64.5
novo_z:65
novo_z+rounded_down:65.5
novo_z:66
novo_z+rounded_down:66.5
novo_z:67
novo_z+rounded_down:67.5
novo_z:68
novo_z+rounded_down:68.5
novo_z:69
novo_z+rounded_down:69.5
novo_z:70
novo_z+rounded_down:70.5
novo_z:71
novo_z+rounded_down:71.5
novo_z:72
novo_z+rounded_down:72.5
novo_z:73
novo_z+rounded_down:73.5

O F:\Mestrado UFF dissertação 2021\Reconstrucao 3D-Versão final Testes\x64\Release\Reconstrução 3D - Douglas Gomes.exe (
processo 848) foi encerrado com o código 0.
Pressione qualquer tecla para fechar esta janela...
```

Figura 49: Console 4 de depuração Visual Studio
Fonte: Autoria própria (2021)

Quando a reconstrução é finalizada, o Visual Studio gera um arquivo de saída com extensão .dat, cujo nome é editado a rigor do próprio usuário. Porém, como dito anteriormente, para visualizar a reconstrução tridimensional,

é necessário a utilização das ferramentas de visualização gráfica, TecPlot e ParaView, para que seja possível analisar o volume gerado. Então, na figura 50, é mostrado o resultado da reconstrução 3D do paralelepípedo de alumínio envolvido em resina com forma de tarugo utilizando o TecPlot.

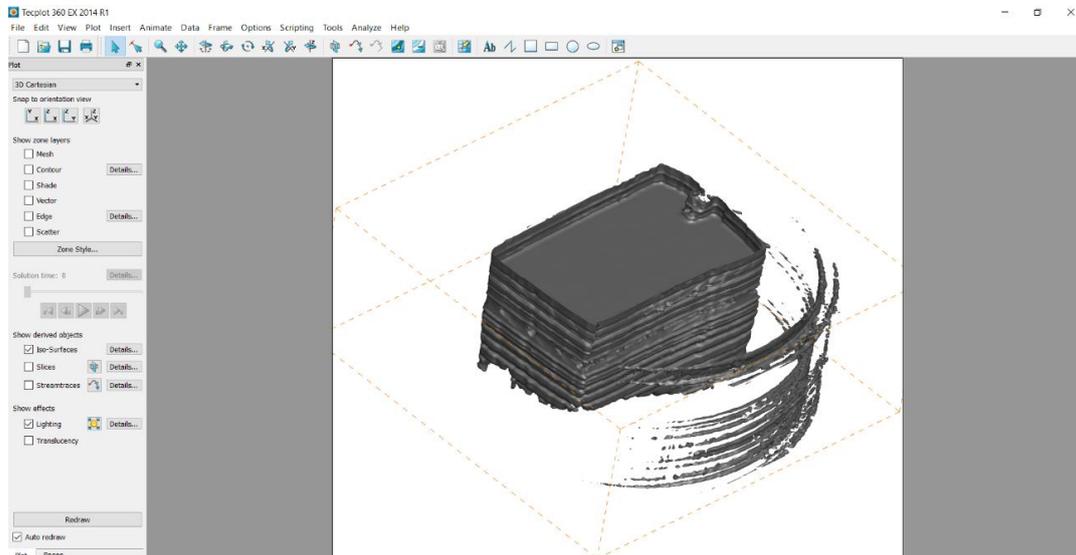


Figura 50: Reconstrução 3D do paralelepípedo de alumínio envolvido em resina - TecPlot

Fonte: Autoria própria (2021)

Com o mesmo arquivo .dat, utilizado no TecPlot, imprime-se o resultado da reconstrução tridimensional utilizando o ParaView. Observa-se na figura 51 o volume gerado com enfoque no paralelepípedo de alumínio.

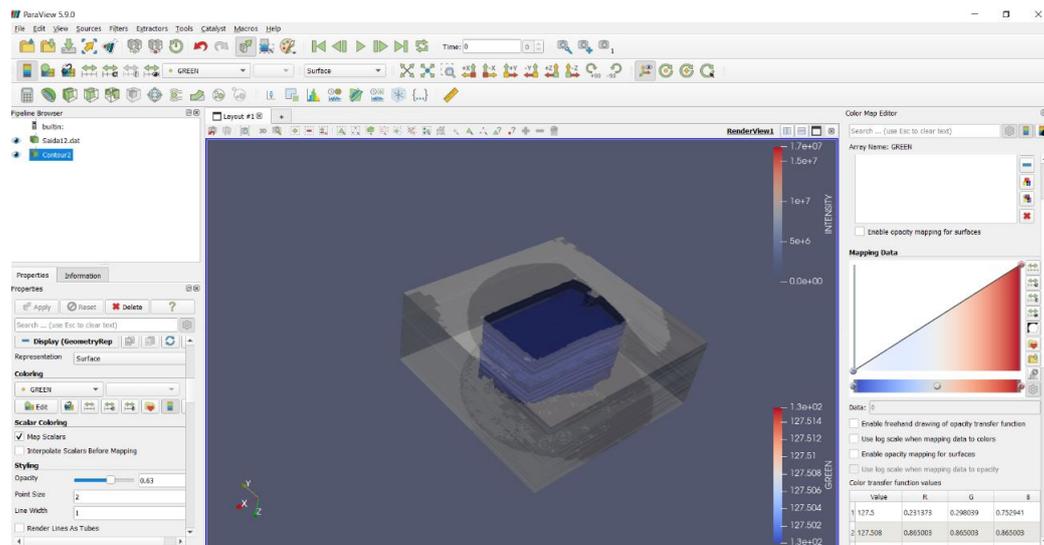


Figura 51: Reconstrução 3D do paralelepípedo de alumínio envolvido em resina - ParaView

Fonte: Autoria própria (2021)

4.2.2 Validação do algoritmo proposto – paralelepípedo de alumínio envolvido em resina

Para garantir que o código está funcionando conforme esperado, assim como na amostra de ferro fundido, utilizou-se o ImageJ para comprovar que o algoritmo é capaz de fazer a reconstrução 3D de forma confiável. Ressalta-se que o ImageJ é um software largamente utilizado em várias literaturas e por esse motivo está sendo considerado como parâmetro para a validação do algoritmo proposto. Assim, segue na figura 52 o início do processo de carregamento das 74 imagens.

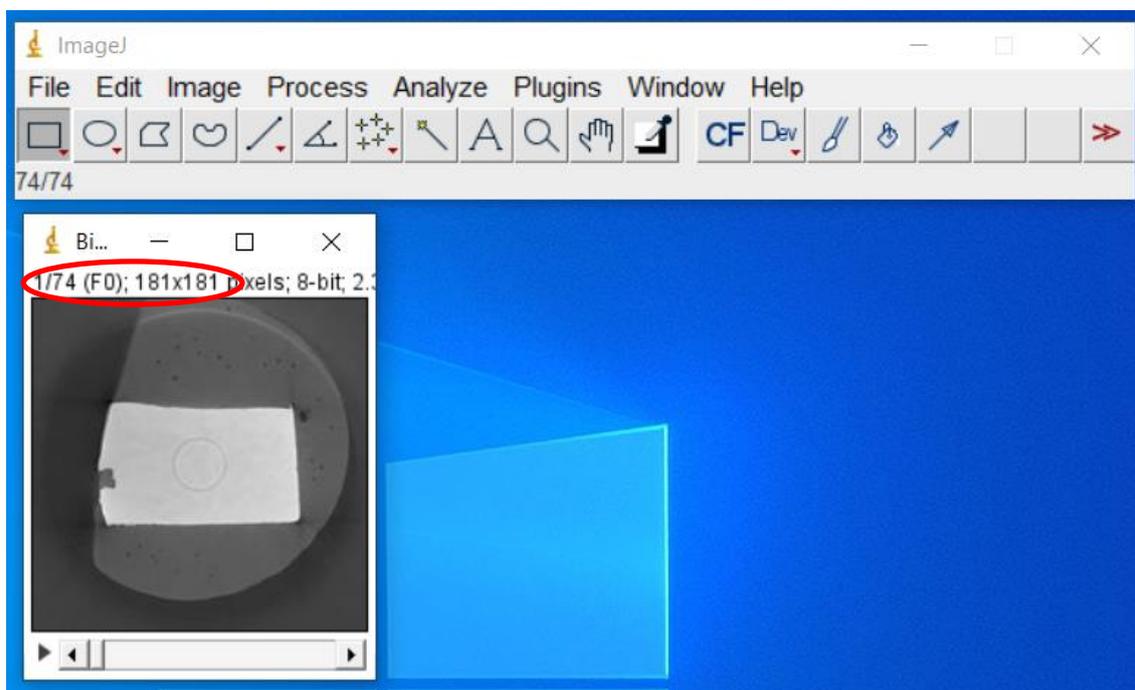


Figura 52: Carregamento das 201 imagens
Fonte: Autoria própria (2021)

Uma diferença do ImageJ em relação ao algoritmo proposto é que no ImageJ todas as imagens selecionadas são carregadas e impressas na tela, indicando ao usuário o quantitativo de arquivos trabalhados. Ressalta-se que o Visual Studio primeiramente gera um arquivo com extensão .dat, já com as características tridimensionais, e depois é necessário a utilização de softwares de visualização gráfica, que no presente trabalho é o TecPlot e ParaView.

Ressalta-se que as imagens carregadas têm uma matriz de 181 x 181 pixels e que foi necessário diminuir o tamanho do arquivo de imagem por conta da limitação do computador, pois quanto maior a resolução maior será o arquivo de trabalho e conseqüentemente é exigido um elevado poder de

processamento. As imagens iniciais, oriundas do V-tomex-m, equipamento responsável pelo seccionamento em série do material de trabalho de forma não destrutiva, possuem em cada imagem uma matriz de 2014 x 2014 pixels.

Então, para que seja feita a reconstrução 3D, é necessário mais um passo que é a geração do volume, dessa forma, segue na figura 53 o passo a passo.

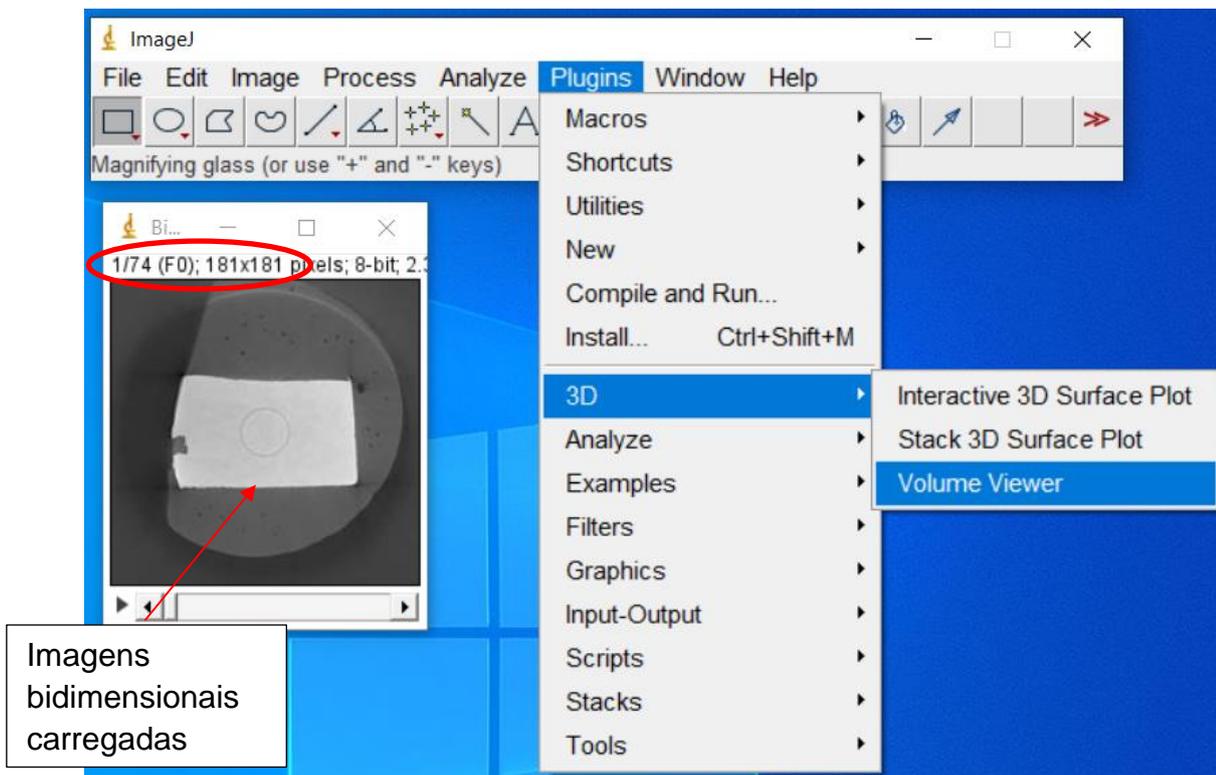


Figura 53: Geração do volume – paralelepípedo de alumínio envolvido em resina

Fonte: Autoria própria (2021)

Assim, para comprovar que o algoritmo é confiável, observa-se na figura 54 que o volume gerado pelo ImageJ é igual aos volumes demonstrados nas ferramentas de visualização (TecPlot e ParaView). Ressalta-se que essas ferramentas de visualização são capazes de fazer a leitura de outras extensões de arquivos, porém para o presente trabalho, utilizou-se a extensão .dat que foi disponibilizada pelo Visual Studio.

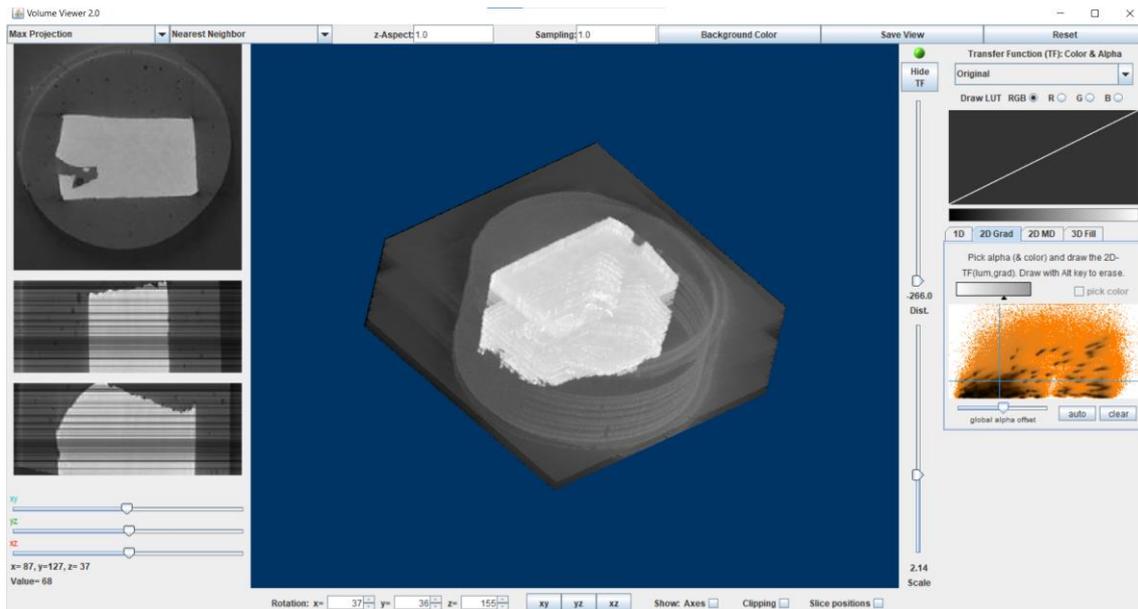


Figura 54: Volume gerado no ImageJ (parte 1) – paralelepípedo de alumínio envolvido em resina
Fonte: Autoria própria (2021)

Dessa forma, para melhorar a diferenciação dos dois tipos de materiais que foram utilizados no seccionamento em série não destrutivo, explorou-se a ferramenta de cores do ImageJ para facilitar a visualização. A figura 55 demonstra o volume conforme comentado.

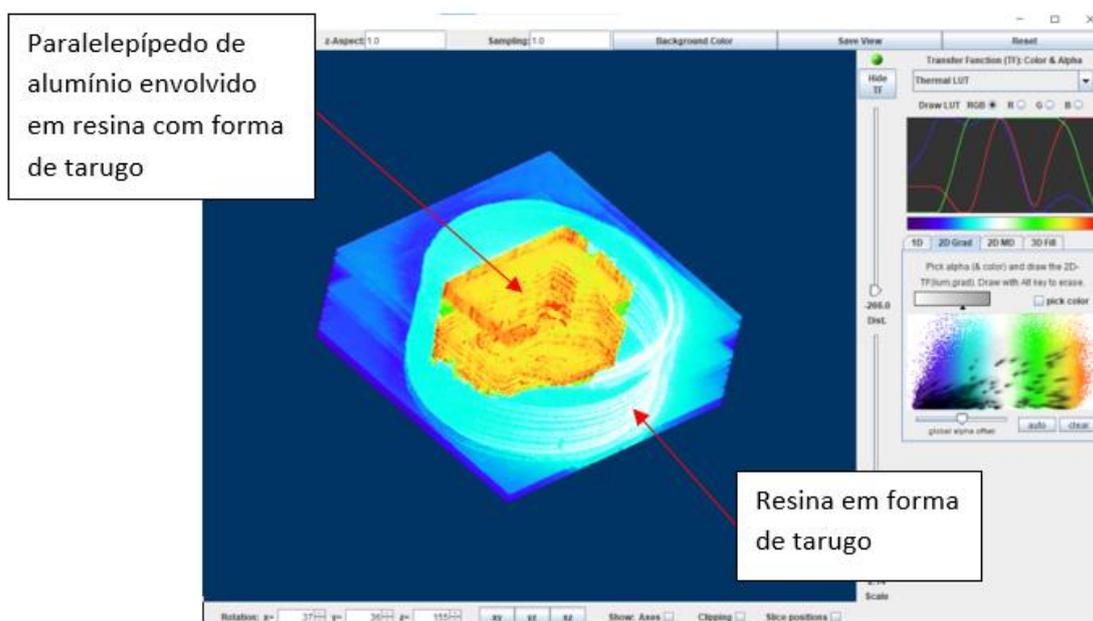


Figura 55: Volume gerado no ImageJ (parte 2) – paralelepípedo de alumínio envolvido em resina
Fonte: Autoria própria (2021)

Outra forma de comparar a reconstrução 3D do paralelepípedo de alumínio envolto em resina, além do ImageJ, foi a captura de uma imagem real, disponibilizada pelo Wysllan Garção, no qual foi possível observar todos as imagens em 3D visualizadas através dos programas de interface gráfica. Assim, para uma melhor visualização e comparação, segue abaixo na figura 56, a visualização do paralelepípedo de alumínio envolvido em resina.

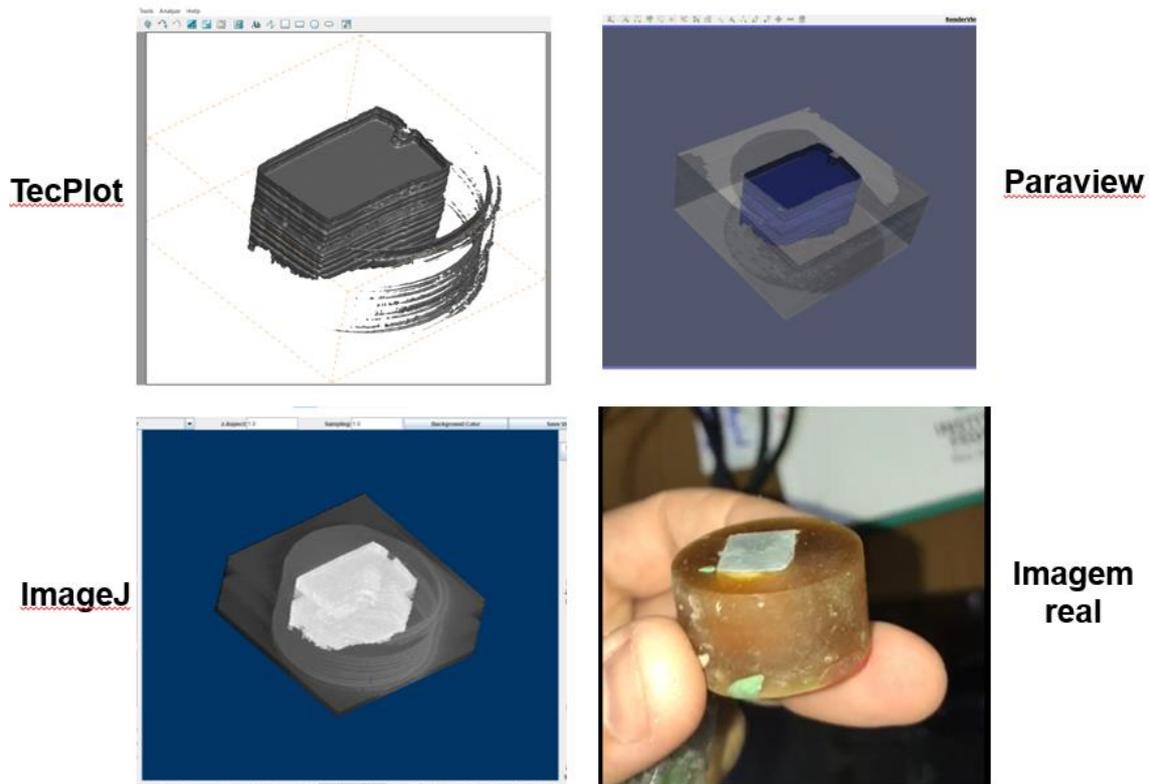


Figura 56: Paralelepípedo de alumínio – TecPlot, ParaView, ImageJ e imagem real
Fonte: Autoria própria (2021)

5. CONCLUSÃO

Como mostrado, de acordo com a pesquisa realizada, os objetivos do trabalho foram alcançados. É notório que a atual versão utilizada para aplicar a reconstrução 3D nas imagens oriundas do seccionamento em série funciona conforme esperado. As ferramentas de visualização, TecPlot e ParaView, em conjunto com o Visual Studio, possibilitaram uma visualização gráfica de forma fidedigna, propiciando uma comparação direta com o ImageJ para a validação do sistema.

O objetivo geral do trabalho foi alcançado, visto que, foi aplicado e validado um código capaz de gerar imagens em 3D que represente materiais do mundo físico com fidelidade através de imagens em 2D extraídas do processo de seccionamento.

Os objetivos específicos também foram alcançados, pois explorou-se a visualização dos volumes através de ferramentas gráficas, salientando que sem elas não seria possível aplicar e validar o algoritmo. As imagens obtidas foram tratadas na entrada de dados inicial, fornecendo informações a classe de dados. As duas reconstruções tridimensionais foram validadas pelo ImageJ, obtendo um resultado satisfatório e comprovando que o programa é confiável. Os dados do domínio discreto foram salvos para que seja possível trabalhar e/ou modificar a reconstrução 3D ou até mesmo realizar novas aplicações em outros tipos de materiais.

Dessa forma, destaca-se que o sistema como um todo é de baixo custo, pois foi totalmente desenvolvido pelo Núcleo de Modelagem Microestrutural da UFF, ressaltando que é um programa *free* no qual também permite a modificação a depender da aplicação, pois a linguagem de programação do código aplicada para a reconstrução tridimensional é aberta.

Um detalhe muito importante foi a parceria entre integrantes externos, internacional e de outra instituição nacional (UFRJ), proporcionando uma sinergia que possibilitou a aplicação prática do algoritmo em duas diferentes técnicas de seccionamento em série, destrutiva e não destrutiva, utilizando dois

equipamentos totalmente diferentes, porém com o mesmo objetivo, gerar imagens 2D para serem reconstruídas de forma tridimensional.

6. TRABALHOS FUTUROS

- ✓ Desenvolver uma interface amigável a fim de facilitar a entrada de dados e a utilização da ferramenta pelos usuários;
- ✓ Implementar a paralelização no código para o processamento das imagens, a fim de torná-lo mais rápido;
- ✓ Implementar um algoritmo que faça a reconstrução tridimensional e possibilite a visualização gráfica no mesmo software;
- ✓ Implementar ferramentas de animações gráficas para facilitar as análises metalográficas;
- ✓ Desenvolver ferramentas para possibilitar que seja feito o tratamento das imagens no mesmo algoritmo a fim de eliminar eventuais impurezas presentes na microestrutura;
- ✓ Desenvolvimento de ferramentas para fazer a binarização e alinhamento das imagens no mesmo programa.

7. REFERÊNCIAS BIBLIOGRÁFICAS

ALKEMPER, J., VOORHEES, P., W. **Quantitative Serial Sectioning Analysis**. [S. l.], 2001.

AMARO JÚNIOR, Edson., YAMASHITA, Helio. **Aspectos básicos de tomografia computadorizada e ressonância magnética**. Revista Brasileira de Psiquiatria, [S. l.], v. 23, n. suppl 1, p. 2–3, 2001. DOI: 10.1590/s1516-44462001000500002.

ANDRADE, Doherty. **Superfícies Parametrizadas**. [S. l.], v. 3, p. 1–8, [s.d.].

ARAÚJO, Olga Maria Oliveira de. **Metodologia para análise de rochas por meio digital utilizando microct e simulações de fluxo de fluidos**. Paper Knowledge . Toward a Media History of Documents, [S. l.], p. 12–26, 2020.

ASSIS, Wesley Luiz da Silva, et al. **Estudo da Influência da Distância entre Camadas na Técnica de Reconstrução 3D por Seção Serial**. Journal of Chemical Information and Modeling, [S. l.], v. 53, n. 9, p. 1689–1699, 2019. DOI: 10.1017/CBO9781107415324.004.

AZEVEDO, Eduardo., CONCI, Aura. **Computação Gráfica Teoria e Prática**. Organization, [S. l.], 2003.

BUENO, Mike. A Cor e a Visão Humana. **Food Chemistry**, [S. l.], v. 53, n. 3, p. 36–41, 2010. DOI: 10.1016/0308-8146(87)90147-6.

CERQUEIRA, Fernando Almeida., NISHIKAWA, Arthur Seiji., GUESSER, Wilson Luiz., AZEVEDO, Cesar Roberto de Farias. **Three-dimensional reconstruction of compacted graphite in vermicular cast iron by manual serial sectioning**. Revista Escola de Minas, [S. l.], v. 68, n. 3, p. 307–312, 2015. DOI: 10.1590/0370-44672014680120.

CHIAVERINI, Vicente. **Aços e Ferros Fundidos**, 6 ed. Publicação da Associação Brasileira de Metalurgia e Materiais, São Paulo, 1990.

COSTA, V. **Ferros Fundidos - Definição Fatores de Influência na Microestrutura do Ferro Fundido**. [S. l.], p. 17–37, 2003.

DEDAVID, BERENICE ANINA., GOMES, CARMEM ISSE., MACHADO, GIOVANNA. **Microscopia Eletrônica de Varredura - Aplicações e Preparação de Amostras - Materiais Poliméricos, Metálicos e Semicondutores**. Dados Internacionais de Catalogação na Publicação (CIP), [S. l.], p. 60, 2007. DOI: 10.1017/CBO9781107415324.004.

DEHOFF, ROBERTO T. **Quantitative Serial Sectioning Analysis: Preview**. [S. l.], v. 131, n. September, p. 259–263, 1983.

DEITEL, PAUL., DEITEL, HARVEY. **How to Program**. [s.l: s.n.]. DOI: 10.1007/978-1-349-07220-0_3.

ESQUEF, ISRAEL ANDRADE. **Técnicas de Entropia em Processamento de Imagens.** [S. l.], 2002.

FARIAS, THIAGO SOUTO MAIOR CORDEIRO DE. **Metodologia para Reconstrução 3D Baseada em Imagens.** [S. l.], 2012.

FELICIANO, PATRÍCIA., LAMEGO, GUILHERME CESAR. **Linguagem de Programação C++.** [S. l.], 2008.

GONZALES; WOODS. **Digital Image Processing**, 2010.

GRAÇA, ANA BEATRIZ RODRIGUES DE ANDRADE. **Desenvolvimento de Ferramenta Computacional para Criação de Domínios Discretos em 3D via Imagens 2D.** [S. l.], 2019.

IMAGEJ. **Home: ImageJ**, c2021. [s.d.]. Disponível em <<https://imagej.nih.gov/ij/>>. Acesso em: 17 de Out. de 2021.

LEÃO, ALEXANDRE CRUZ., ARAÚJO ARNALDO DE ALBUQUERQUE., SOUZA, LUIZ ANTÔNIO CRUZ. **Implementação de Sistema de Gerenciamento de Cores para Imagens Digitais.** [S. l.], n. May 2014, 2014.

LEITE, FREDERICO NOGUEIRA. **Calibração de Dispositivos a Cores Utilizando uma Camera Digital.** [S. l.], 2006.

LI, M.; GHOSH, S.; RICHMOND, O.; WEILAND, H.; ROUNS, T. N. **Three Dimensional Characterization and Modeling of Particle Reinforced Metal Matrix Composites part II: Damage Characterization.** Materials Science and Engineering A, [S. l.], v. 266, n. 1–2, p. 221–240, 1999. DOI: 10.1016/S0921-5093(98)01133-2.

LIEBERMAN, SCOTT I. **Microstructural Characterization, Visualization and Simulation of Ti-B Materials.** Environmental Engineering, [S. l.], n. May, 2007.

MASSIRER JUNIOR, ERCIO., GUESSER, WILSON LUIZ. **Avaliação da Usinabilidade do Ferro Fundido Vermicular através de Ensaios de Furação.** Manufacturing Engineering, [S. l.], 2011.

MELCHIADES, FÁBIO G., BOSCHI, ANSELMO ORTEGA. **Cores e Tonalidades em Revestimentos Cerâmicos.** Cerâmica Industrial, [S. l.], v. 4, n. December, p. 11–18, 1999. Disponível em: http://www.ceramicaindustrial.org.br/pdf/v04n16/v4n16_2.pdf.

MOREIRA, ADRIANA JOSÉ DA PENHA. **A Microtomografia de Raios X para o Estudo da Anatomia da Microestrutura de Órgãos e suas Patologias.** [S. l.], p. 6, 2021.

OLIVEIRA, ERIVAM MORAIS DE. **Da Fotografia Analógica à Ascensão da fotografia Digital.** BOCC. Biblioteca On-line de Ciências da Comunicação, [S. l.], 2006.

OPENCV. **OpenCV: Home**, c2021. Página inicial. Disponível em <<https://opencv.org/>>. Acesso em: 16 de Out. de 2021.

PARAVIEW. **ParaView**, c2021. Página inicial. Disponível em < <https://www.paraview.org/>>. Acesso em: 16 de Out. de 2021.

PEIXOTO, FILIPE DE MACÊDO., et al. **Desenvolvimento de um Software para Cálculo da Densidade de Nódulos de Grafita em Ferro Fundido Nodular através de Processamento Digital de Imagens**. *Revista Materia*, [S. l.], v. 20, n. 1, p. 262–272, 2015. DOI: 10.1590/S1517-707620150001.0026.

PRAZERES, TALITA FONSECA DOS., ASSIS, WESLEY LUIZ DA SILVA., GRAÇA, ANA BEATRIZ RODRIGUES DE ANDRADE. **Influência da Resolução da Imagem na Reconstrução Microestrutural em 3D por Seccionamento em Série**. [S. l.], n. Xxxviii, p. 3–4, 2018. DOI: 10.9781/iji-mai.2014.261.

RIBEIRO, MARIA DA CONCEIÇÃO SANTOS. **As cores e a Visão e a Visão das Cores**. [S. l.], p. 86, 2011.

ROCHA, JOÃO CARLOS. **Cor Luz, Cor Pigmento e os Sistemas RGB e CMY**. *Revista Belas Artes*, [S. l.], v. 2, n. 3, p. 1–19, 2010.

SCHILDT, HERBERT. **C Completo e Total**. *Makron American Journal of Archaeology*, 2000. DOI: 10.2307/506804.

SCURI, ANTONIO ESCAÑO. **Fundamentos da Imagem Digital**. [S. l.], 1999.

SILVA, LUCIANA FERREIRA. **Aquisição, Processamento e Análise de Imagens 3D: MicroTC e FIB-SEM na Caracterização de Defeitos em Solda Molhada**. [S. l.], p. 151, 2014.

SILVA, TENNYSON RENÉ. **Um Novo Método para a Caracterização de Cálculos Urinários por Tomografia Computadorizada e Técnicas de Metalografia**. [S. l.], 2004.

SOUZA, GABRIELLE COSTA DE. **Ferro Fundido**. [S. l.], p. 1–7, 2019.

TEC PLOT. **TecPlot CFD Post-processing Data Visualization and Analysis**, c2021. Página inicial. Disponível em < <https://www.tecplot.com/>>. Acesso em: 17 de Out. de 2021.

TELES, ÁTILA DE PAIVA. **Aplicação da Microtomografia Computadorizada de Raios X por Dupla-Energia na Caracterização de Materiais Porosos**. *Angewandte Chemie International Edition*, 6(11), 951–952., [S. l.], p. 5–24, 2016.

TEWARI, ASIM; GOKHALE, ARUN M. **Application of Three-Dimensional Digital Image Processing for Reconstruction of Microstructural Volume from Serial Sections**. *Materials Characterization*. [S. l.], v. 44, n. 3, p. 259–269, 2000. a. DOI: 10.1016/S1044-5803(99)00004-2.

UCHIC, MICHAEL D. **Serial Sectioning Methods for Generating 3D Characterization Data of Grain and Precipitate-Scale Microstructures**. *Computational Methods for Microstructure-Property Relationships*, [S. l.], p. 31–52, 2011. DOI: 10.1007/978-1-4419-0643-4.

UCHIC, MICHAEL D., et al. **An Automated Multi-Modal Serial Sectioning System for Characterization of Grain-Scale Microstructures in Engineering Materials**. 1st International Conference on 3D Materials Science 2012, 3DMS 2012, [S. l.], p. 195–202, 2012. DOI: 10.1007/978-3-319-48762-5_30.

UES. **UES, Inc. - Excellence in Science & Technology**, c2021. Página inicial. Disponível em <<https://www.ues.com/>>. Acesso em: 10 de Nov. de 2021.

VIDAL, DOUGLAS FERREIRA. **Análise de Estrutura e Propriedades Mecânicas de um Ferro Fundido Nodular em Processo de Fundição produzido pela Técnica de Imersão de Sino**. [S. l.], p. 1–172, 2013.

VISUAL STUDIO. **Visual Studio: IDE e Editor de Código para Desenvolvedores**, c2021. Página inicial. Disponível em <<https://code.visualstudio.com/>>. Acesso em: 15 de Out. de 2021.

ZANKEL, ARMIN., WAGNER, JULIAN., POELT, PETER. **Serial Sectioning Methods for 3D Investigations in Materials Science**. Micron, [S. l.], v. 62, p. 66–78, 2014. DOI: 10.1016/j.micron.2014.03.002. Disponível em: <http://dx.doi.org/10.1016/j.micron.2014.03.002>.

APÊNDICE A – Tutorial Tecplot

Como um breve tutorial, segue abaixo alguns passos de como abrir um arquivo no Tecplot. Ressalta-se que foi demonstrado o passo a passo utilizando as imagens de ferro fundido, porém para a visualização do volume reconstruído, independentemente do tipo de material, seja ele o paralelepípedo de alumínio envolvido em resina ou o ferro fundido, o processo de manipulação no TecPlot (apêndice A) e/ou ParaView (apêndice B) é o mesmo, pois trata-se de imagens já obtidas e manipuladas nos equipamentos de seccionamento. Então, primeiramente, na figura 57, deve-se importar as imagens através da aba *File*, logo em seguida clicar em *Open Layout*. Outra forma de abrir o arquivo é clicando em *Ctrl + O*.

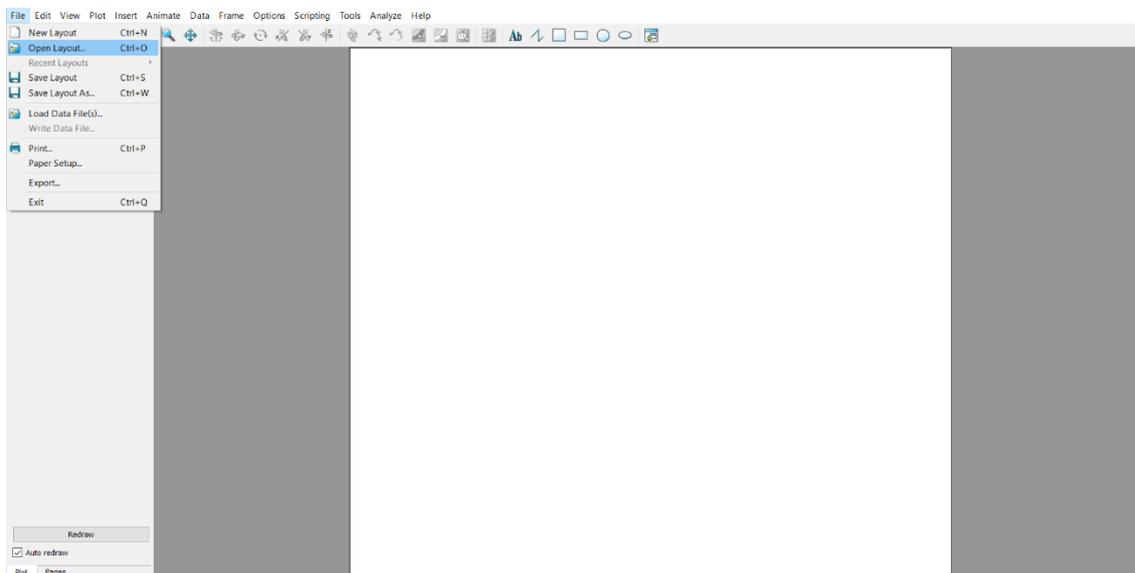


Figura 57: Abrindo o arquivo com a extensão .dat
Fonte: Autoria própria (2021)

Após finalizar o primeiro passo, o Tecplot irá abrir uma janela chamada *Load Data Files* onde indica o diretório no qual o arquivo .dat se encontra. Nele é possível navegar pelas pastas até encontrar o arquivo desejado. Alguns pontos devem ser levados em consideração no momento de abertura do arquivo para que não haja erros no carregamento feito pelo programa. Sendo assim, é necessário informar o tipo de arquivo, ou seja, a extensão na aba *Files of Types* e também o nome do arquivo com a extensão .dat que foi salvo no diretório na aba *File name*. Observe-se na figura 58 esse procedimento de forma simplificada.

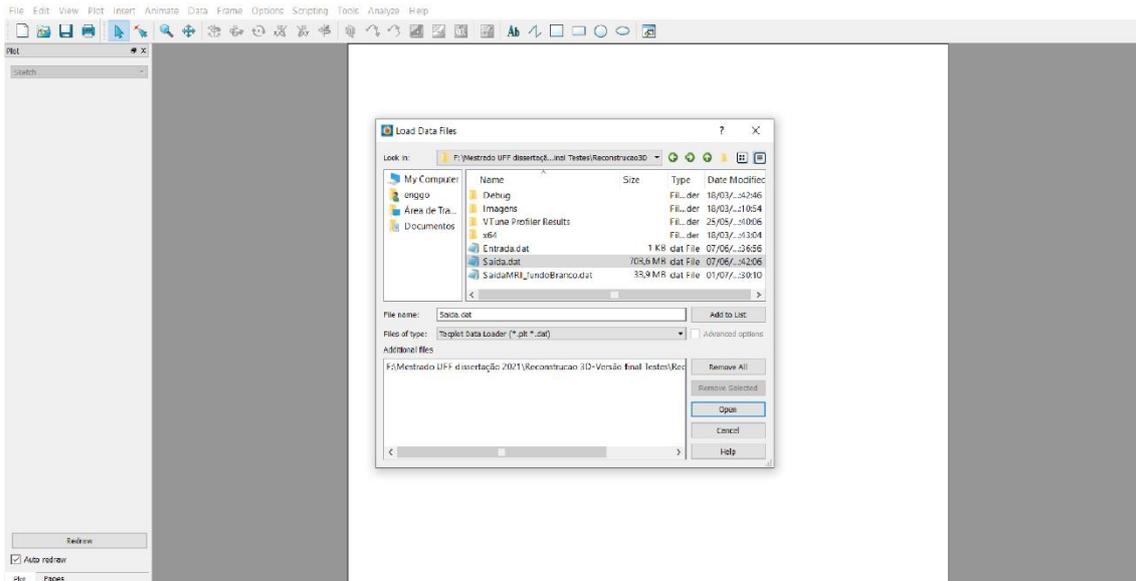


Figura 58: Diretório do arquivo
Fonte: Autoria própria (2021)

Assim que o arquivo .dat for selecionado, basta clicar no botão *Open*. Após esse procedimento, o Tecplot abrirá a microestrutura correspondente ao arquivo de trabalho. Observa-se, na figura 59, o cubo gerado pela reconstrução tridimensional. Como observação, o cubo é o objeto de estudo no qual foi gerado em função da simulação computacional.

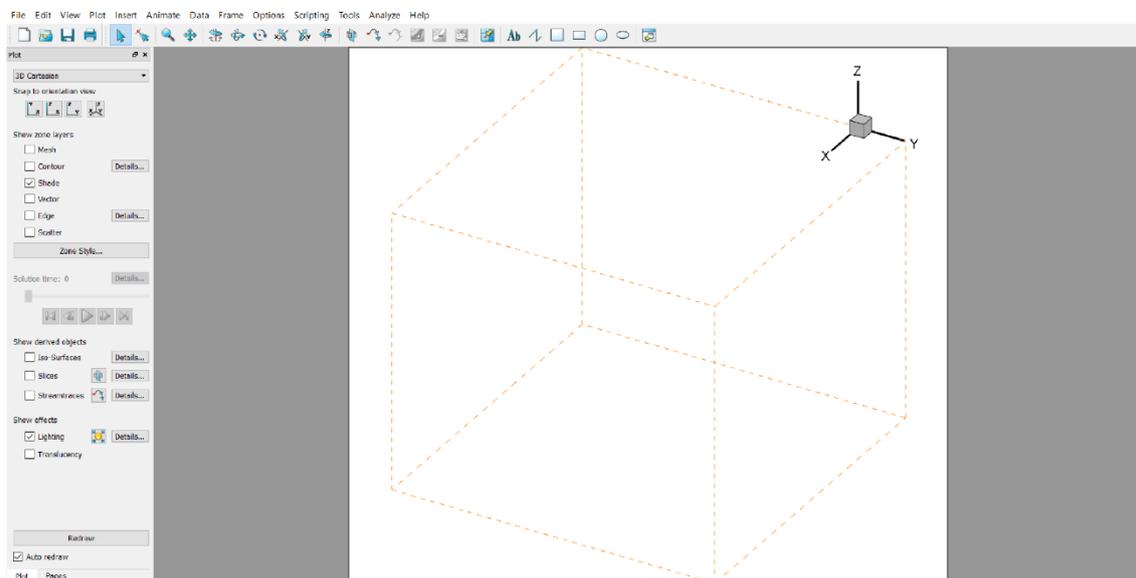


Figura 59: Primeira visualização do arquivo .dat
Fonte: Autoria própria (2021)

Para que seja possível visualizar a microestrutura e conseqüentemente analisar a imagem, necessita-se selecionar a caixa *Contour*. Assim que essa

etapa for cumprida, aparecerá uma mensagem. Basta clicar em “Yes” e prosseguir.

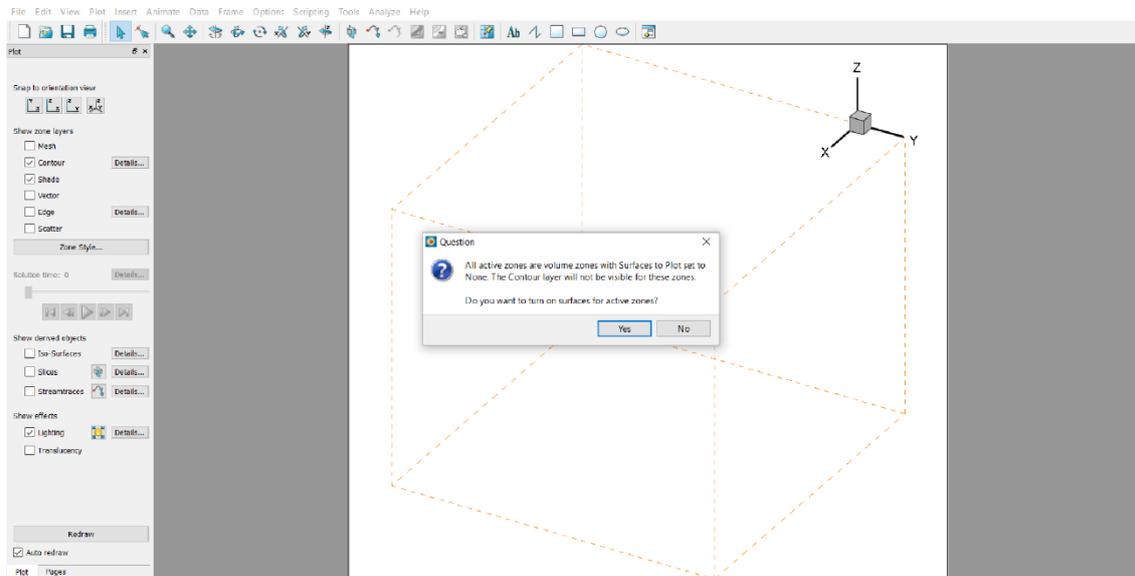


Figura 60: Ativando a caixa “Contour”
Fonte: Autoria própria (2021)

Após clicar no botão “Yes”, nota-se que a microestrutura modificou as cores. Esse processo foi possível por conta da seleção da opção “Contour”. Observa-se na figura 61 o resultado dessa operação.

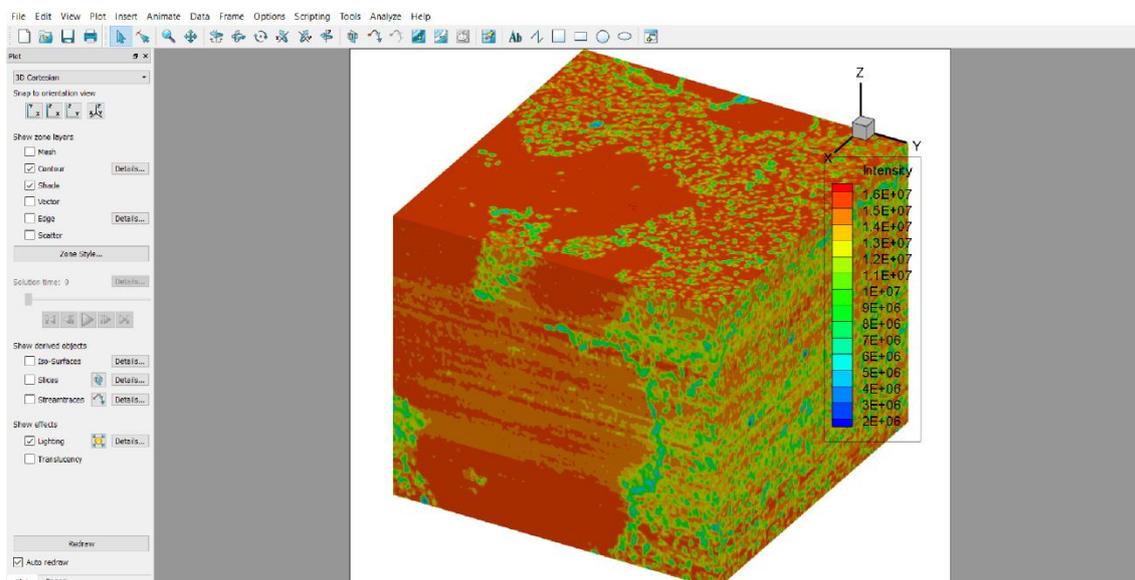


Figura 61: Ativando a caixa “Contour”
Fonte: Autoria própria (2021)

Para que a microestrutura fique em uma posição aproximada a da utilizada pelo *ImageJ*, explorou-se os recursos de rotação para facilitar a

visualização. Como observação, utilizou-se a ferramenta de rotação tanto no *Tecplot* quanto no *ImageJ* para enfatizar o nódulo de ferro fundido no qual foi possível ser visualizado no fundo da microestrutura. Observa-se, na figura 62, que os eixos cartesianos estão invertidos.

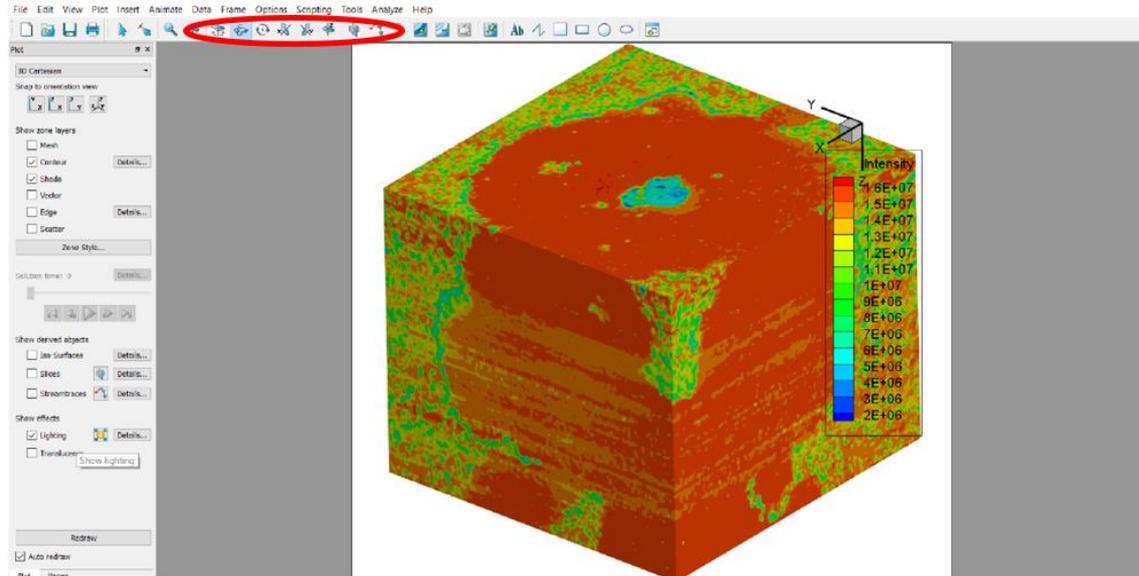


Figura 62: Rotacionando a microestrutura
Fonte: Autoria própria (2021)

Outra opção para facilitar uma análise mais detalhada é a possibilidade de trabalhar com eixos cartesianos em duas ou três dimensões, seccionar a microestrutura e também trabalhar com várias opções de layers que podem ser facilmente visualizados ao lado esquerdo da figura 63.

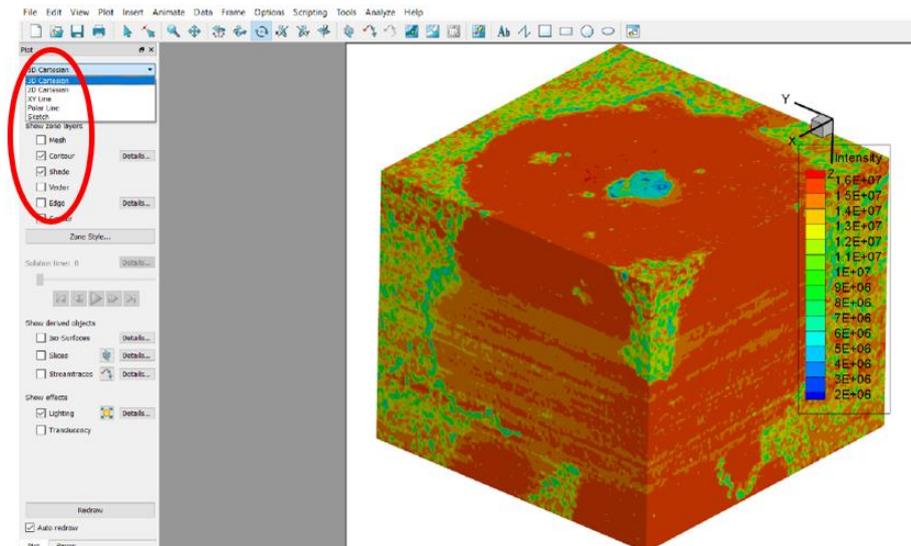


Figura 63: Opções de visualização
Fonte: Autoria própria (2021)

Caso haja a necessidade de abrir o arquivo com a utilização de ferramentas de imagem, como, por exemplo, o visualizador de imagens do Windows, é possível salvar o arquivo trabalhado e posteriormente fazer uma apresentação ou até mesmo compartilhar a análise, através de figuras, com outras pessoas. Segue na figura 64 a opção de armazenamento com extensão jpeg.

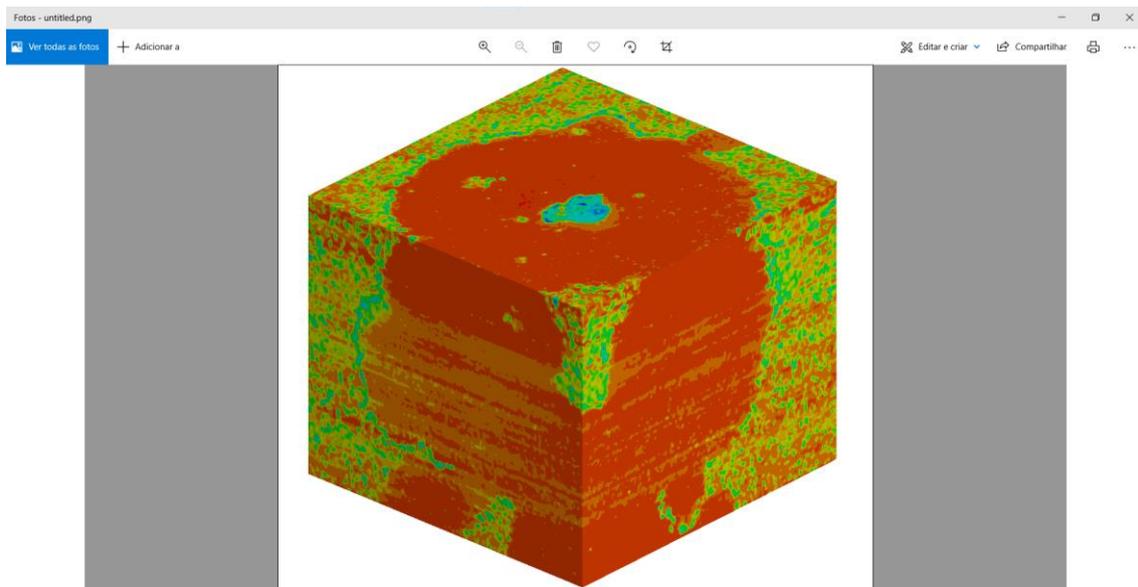


Figura 64: Opções de visualização
Fonte: Autoria própria (2021)

APÊNDICE B – Tutorial ParaView

Também, como um breve tutorial, segue abaixo alguns passos de como abrir e manusear um arquivo .dat no ParaView. Primeiramente, na figura 65, deve-se importar as imagens através da aba *File Open* ou na pasta amarela indicada, logo em seguida, após encontrar o arquivo .dat clicar em *ok*. Outra forma de abrir o arquivo é clicando em *Ctrl + O*.

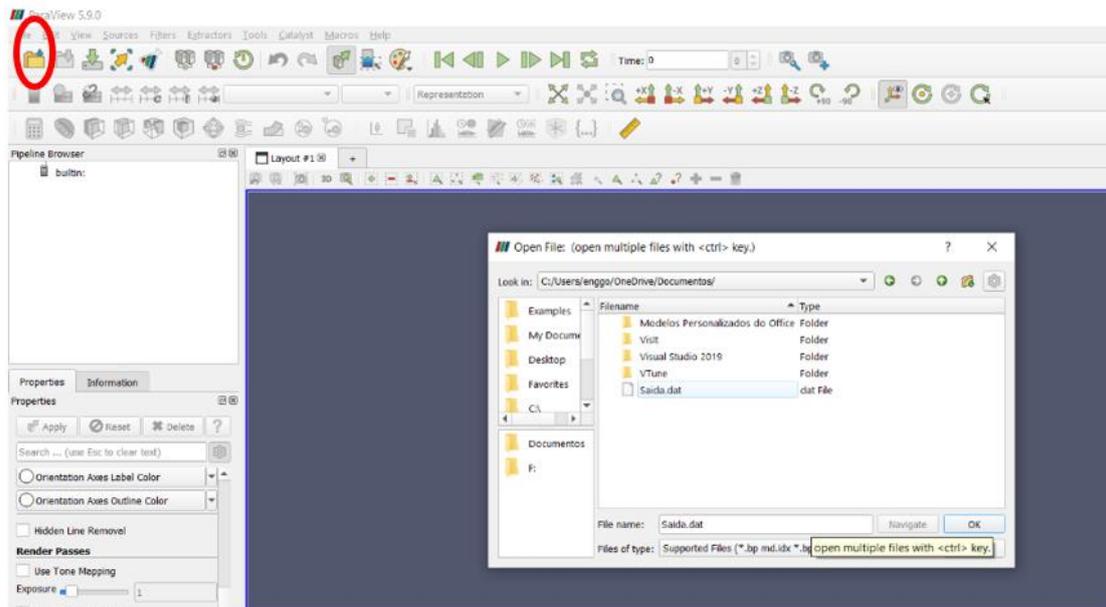


Figura 65: Abrindo o arquivo no ParaView 1
Fonte: Autoria própria (2021)

Após esse primeiro passo, o programa irá imprimir na tela, conforme figura 66, as opções de abertura do arquivo. Nesse caso, para o presente trabalho foi utilizado o *Tecplot Reader*. Feito isso, basta clicar em *ok*.

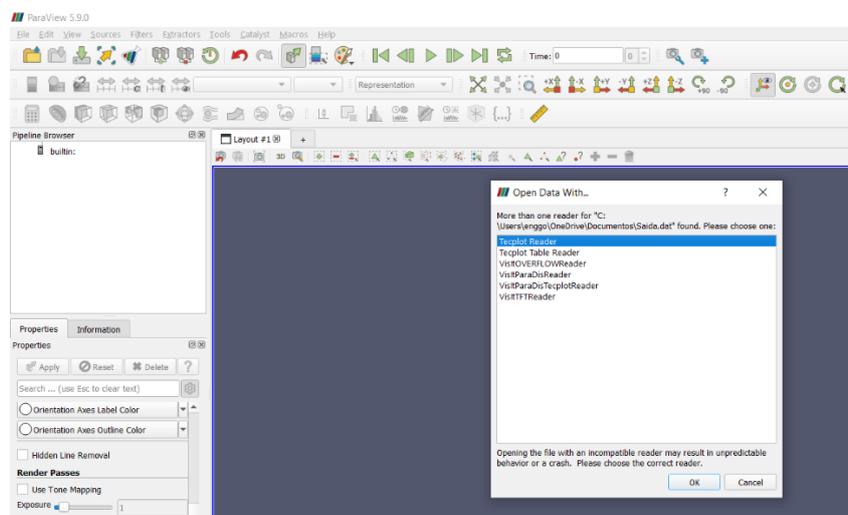


Figura 66: Abrindo o arquivo no ParaView 2
Fonte: Autoria própria (2021)

Quando o arquivo é aberto, no canto superior esquerdo da figura 67, aparecerá o nome do arquivo que o programa está lendo. Porém, pode acontecer do “olho”, onde encontra-se o nome do arquivo que nesse caso é Saída.dat, estar fechado. Então, para que o arquivo possa aparecer corretamente, basta dar um clique com o botão esquerdo em cima do “olho” que a imagem irá aparecer na tela principal.

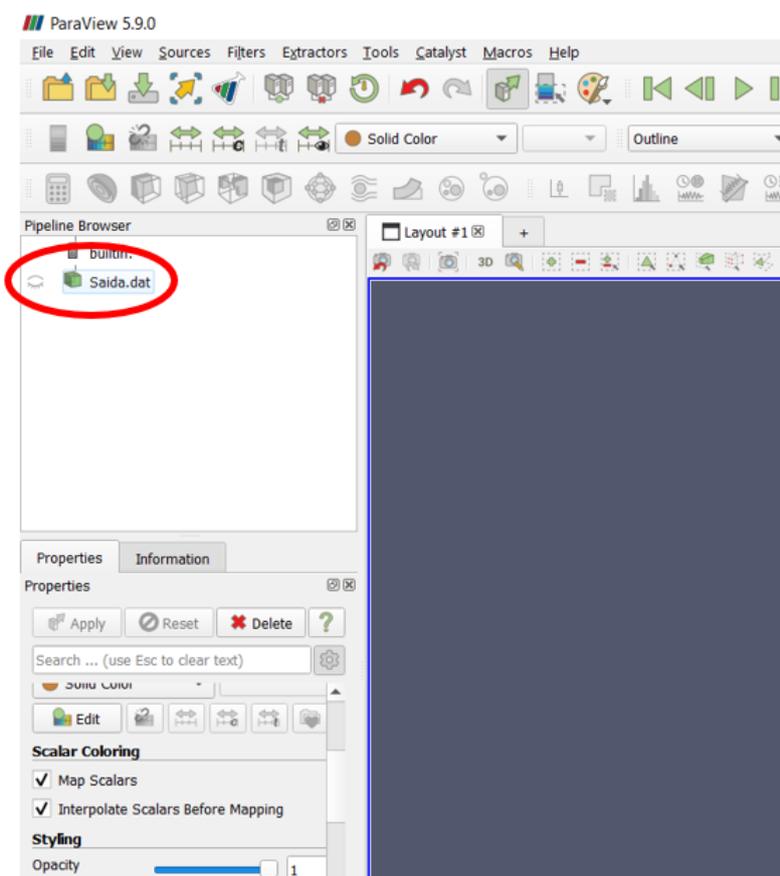


Figura 67: Visualização do arquivo
Fonte: Autoria própria (2021)

Após realizar o procedimento anterior, o arquivo automaticamente aparecerá na tela. Observa-se na figura 68 que a imagem está sendo impressa na parte superior da microestrutura, então, para facilitar a visualização do volume, basta utilizar as ferramentas de rotação do ParaView.

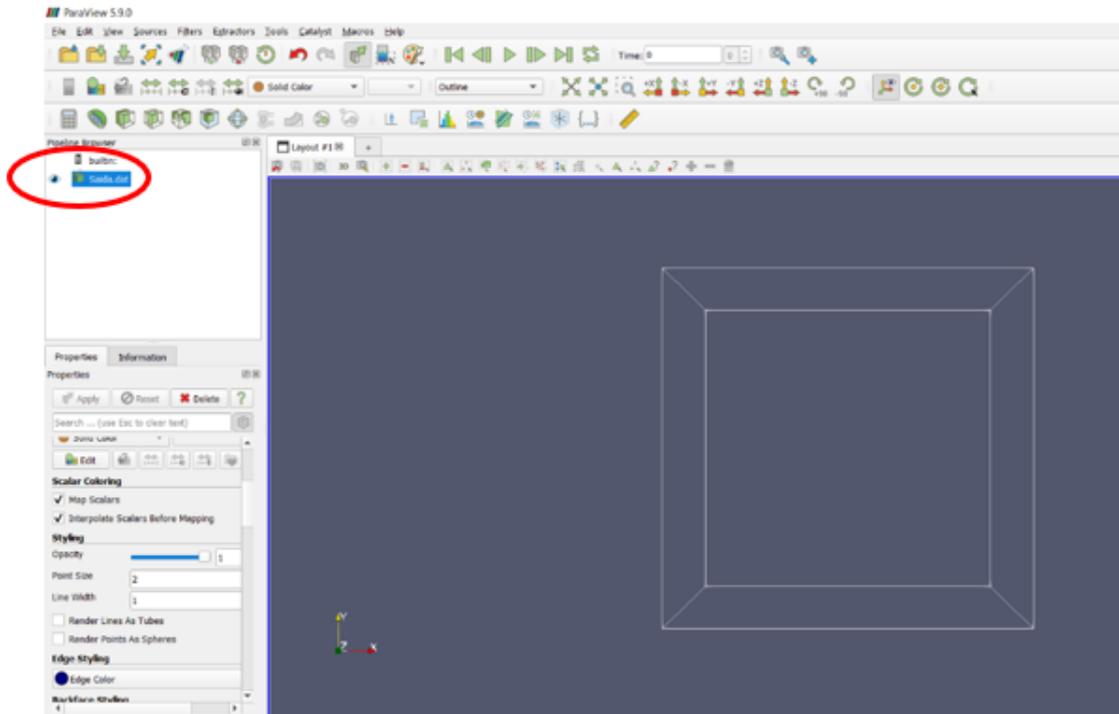


Figura 68: Arquivo impresso na tela – visão de topo
Fonte: Autoria própria (2021)

Como exemplo, após a utilização da ferramenta de rotação, segue na figura 69 o arquivo inicial com uma visualização tridimensional.

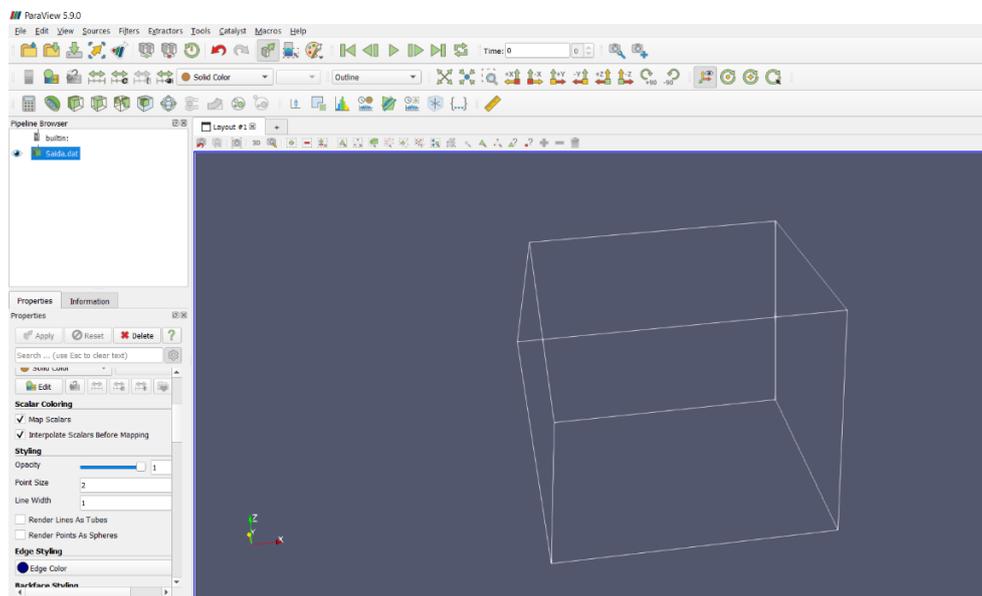


Figura 69: Arquivo impresso na tela – visão tridimensional
Fonte: Autoria própria (2021)

Dessa forma, para facilitar a visualização da microestrutura e possibilitar a análise com maiores detalhes, modifica-se a opção indicada na figura 70 para *surface*.

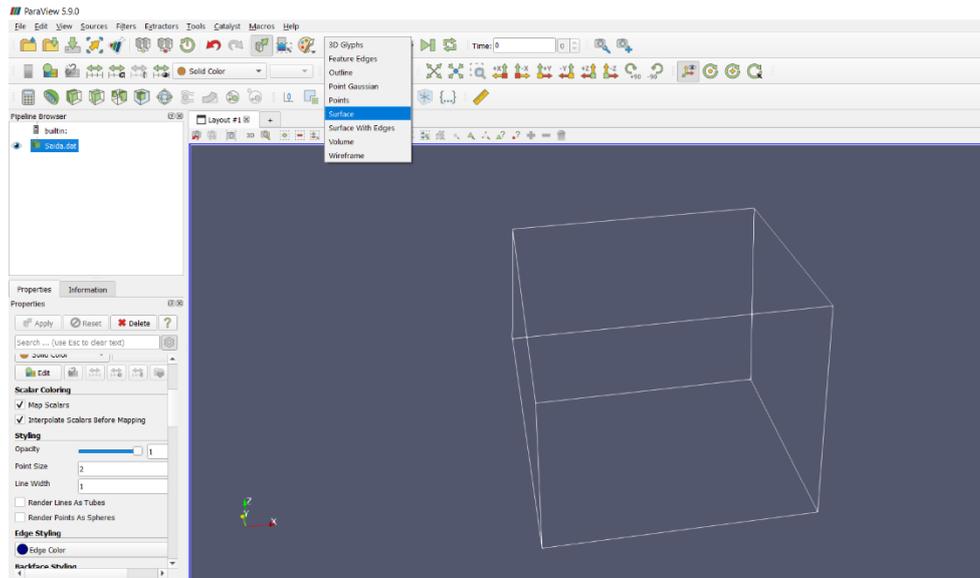


Figura 70: Utilização de ferramentas de rotação
Fonte: Autoria própria (2021)

Feita essa modificação para *surface*, o volume se torna totalmente sólido, não possibilitando a análise detalhada da microestrutura pois está no modo *Solid Color*. Então, para que seja possível analisar com maiores detalhes os grãos de ferro fundido nodular, basta trocar, nas opções de cores, para *INTENSITY*. Na figura 71 é demonstrado esse passo.

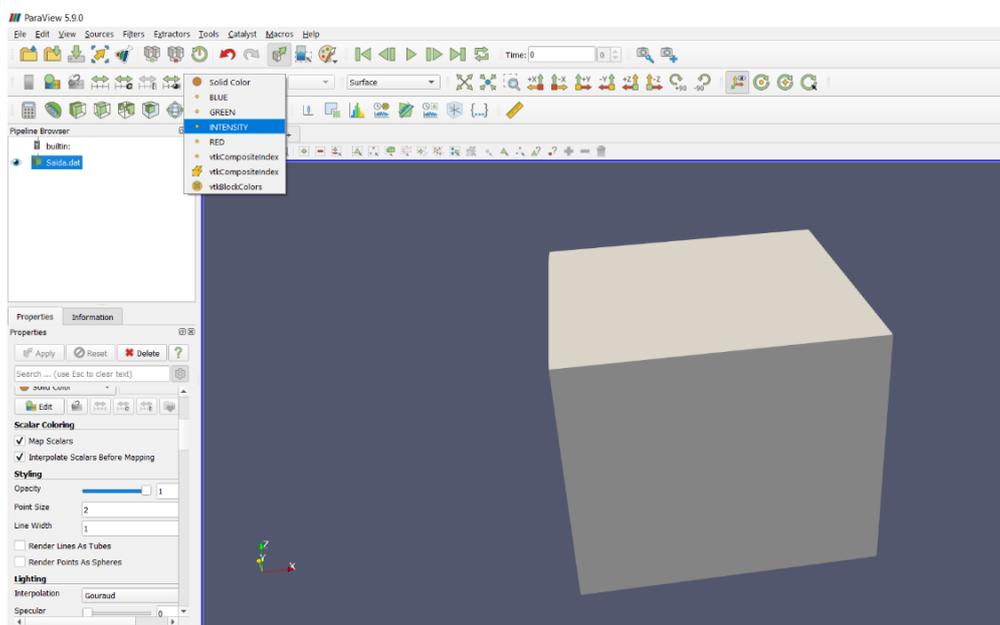


Figura 71: Opções de análise
Fonte: Autoria própria (2021)

Após a realização dos procedimentos anteriores, é possível visualizar a microestrutura com uma maior riqueza de detalhes, facilitando assim a sua

análise por profissionais de metalografia. Segue na figura 72 o volume trabalhado.

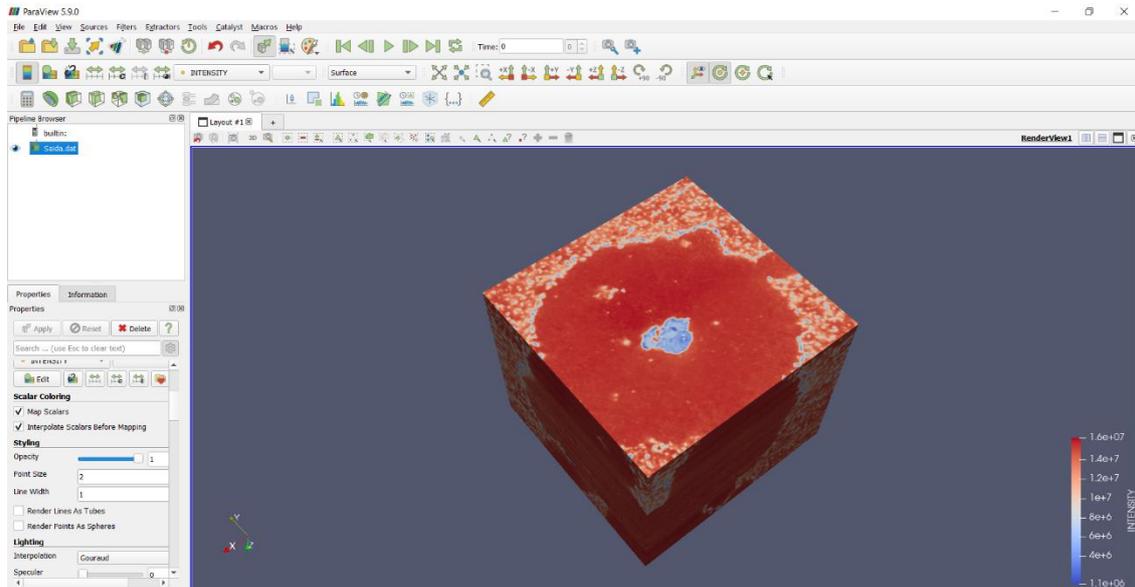


Figura 72: Volume detalhado
Fonte: Autoria própria (2021)

Outra opção muito interessante é a ferramenta de contorno. Com essa opção do ParaView, é possível visualizar os grãos de ferro fundido nodular de forma separada do volume como um todo, facilitando e muito as análises. Para acessar essa ferramenta, basta clicar no botão *Contour* que automaticamente aparecerá a opção abaixo do arquivo de Saída.dat. Observa-se na figura 73 a demonstração dessa opção.

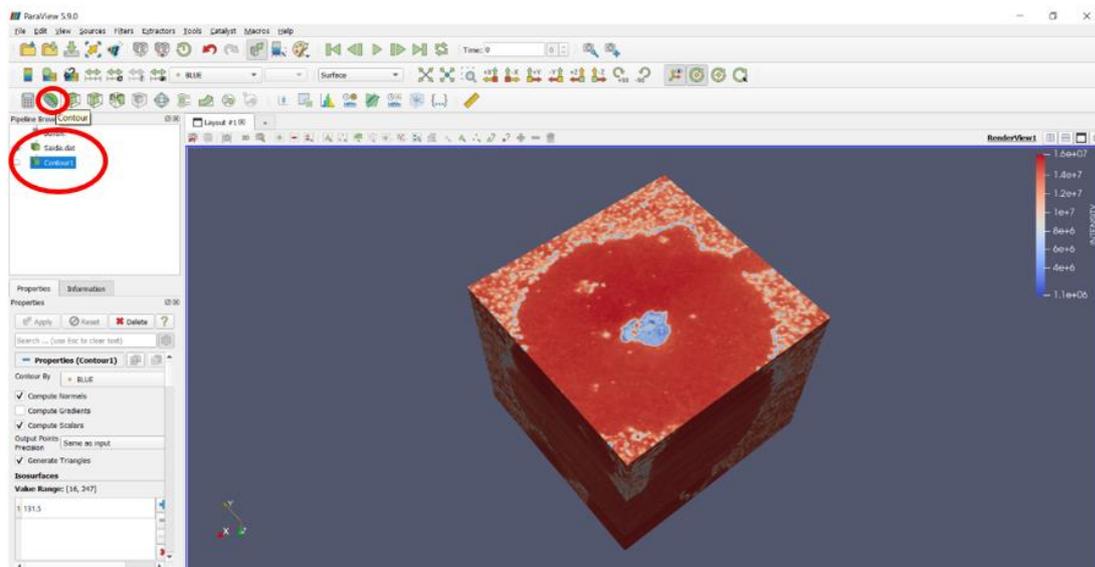


Figura 73: Ferramentas de contorno
Fonte: Autoria própria (2021)

Dessa forma, após a utilização dessa opção, é impresso na tela apenas os grãos, excluindo os contornos da microestrutura. Segue demonstração na figura 74.

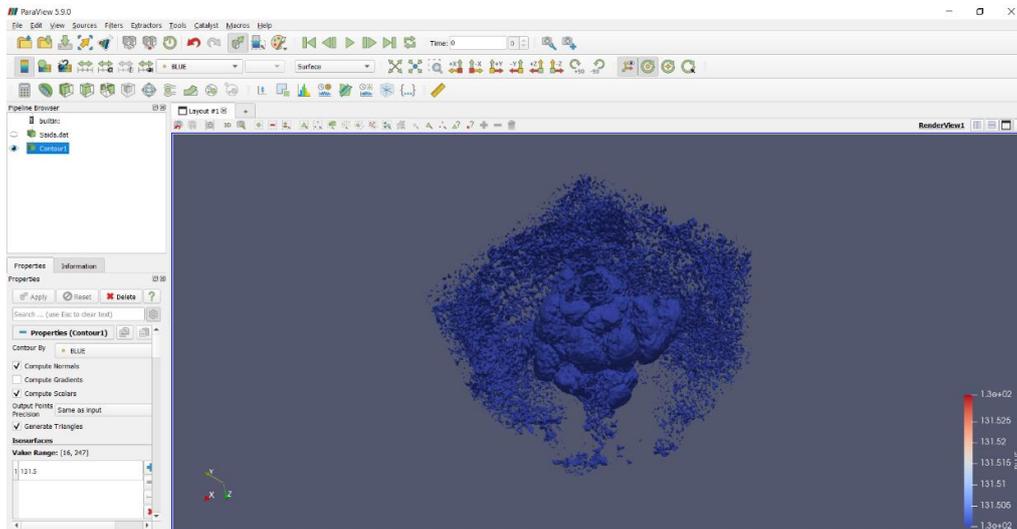


Figura 74: Arquivo com utilização da ferramenta de contorno
Fonte: Autoria própria (2021)

Caso haja a necessidade de abrir o arquivo com a utilização de ferramentas de imagem, como, por exemplo, o visualizador de imagens do Windows, é possível salvar o arquivo trabalhado e posteriormente fazer uma apresentação ou até mesmo compartilhar a análise, através de figuras, com outras pessoas. Segue na figura 75 a opção de armazenamento.

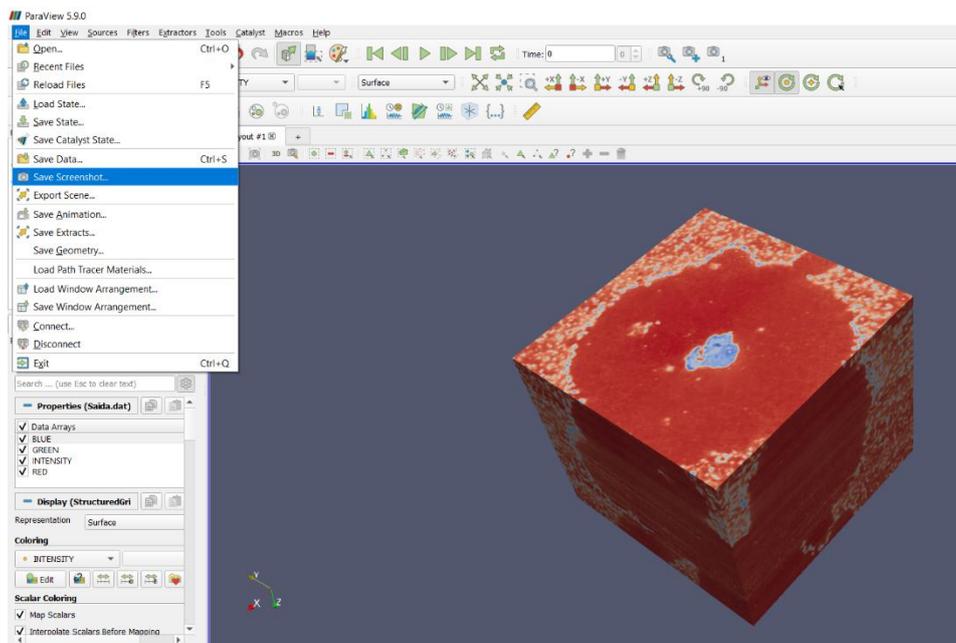


Figura 75: Opção para salvar a microestrutura em arquivo de imagem
Fonte: Autoria própria (2021)

Além da opção de salvar a microestrutura desenvolvida em arquivo de imagem, o ParaView dispõe da possibilidade de escolher o tipo de extensão de imagem que o usuário preferir, sendo que é disponibilizado pelo programa as extensões .bmp, .jpg ou .jpeg, .png e .tif. Segue na figura 76 o demonstrativo citado.

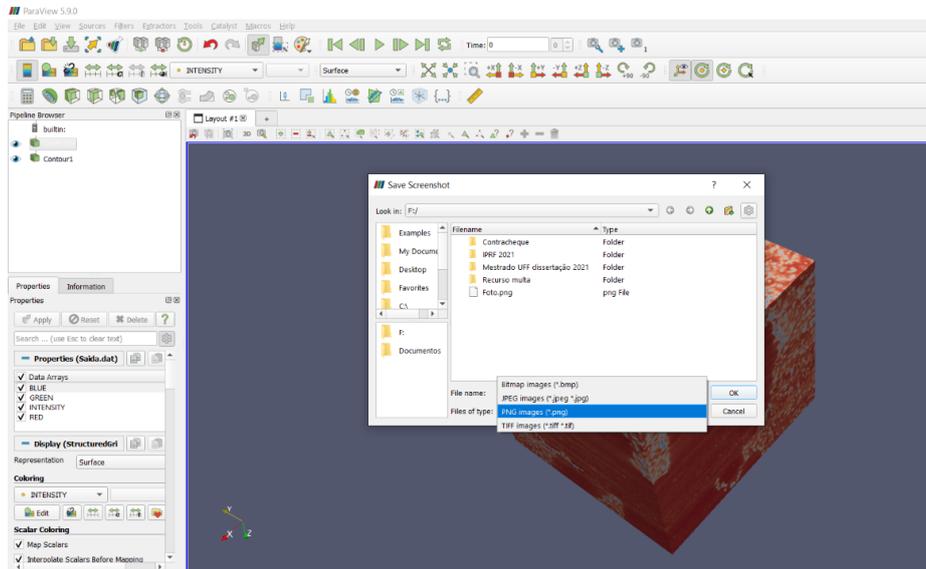


Figura 76: Abrindo o arquivo no ParaView
Fonte: Autoria própria (2021)

Pode-se observar, na figura 77, a microestrutura utilizada no presente trabalho salva como uma imagem com extensão .png, que é facilmente lida em programas de leitura de imagens, que nesse caso foi o visualizador de fotos do Windows.

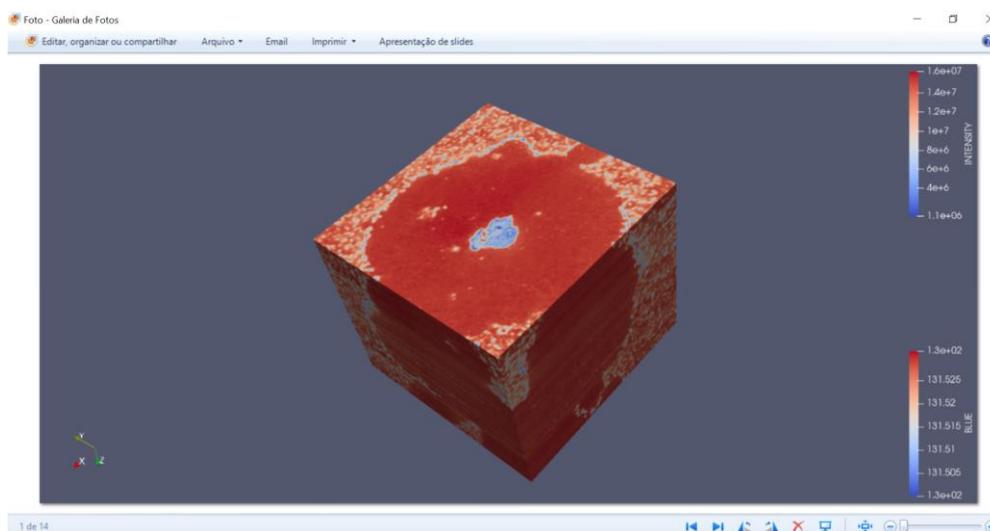


Figura 77: Arquivo de imagem
Fonte: Autoria própria (2021)

APÊNDICE C – Tutorial *ImageJ*

Como um breve tutorial, segue abaixo alguns passos para o desenvolvimento de uma simulação utilizando o ImageJ, iniciando desde a abertura das imagens até a reconstrução tridimensional propriamente dita.

Primeiramente, na figura 78, deve-se importar as imagens através da aba *File*, logo em seguida em *Import*, e depois selecionar a aba *Image Sequence*.

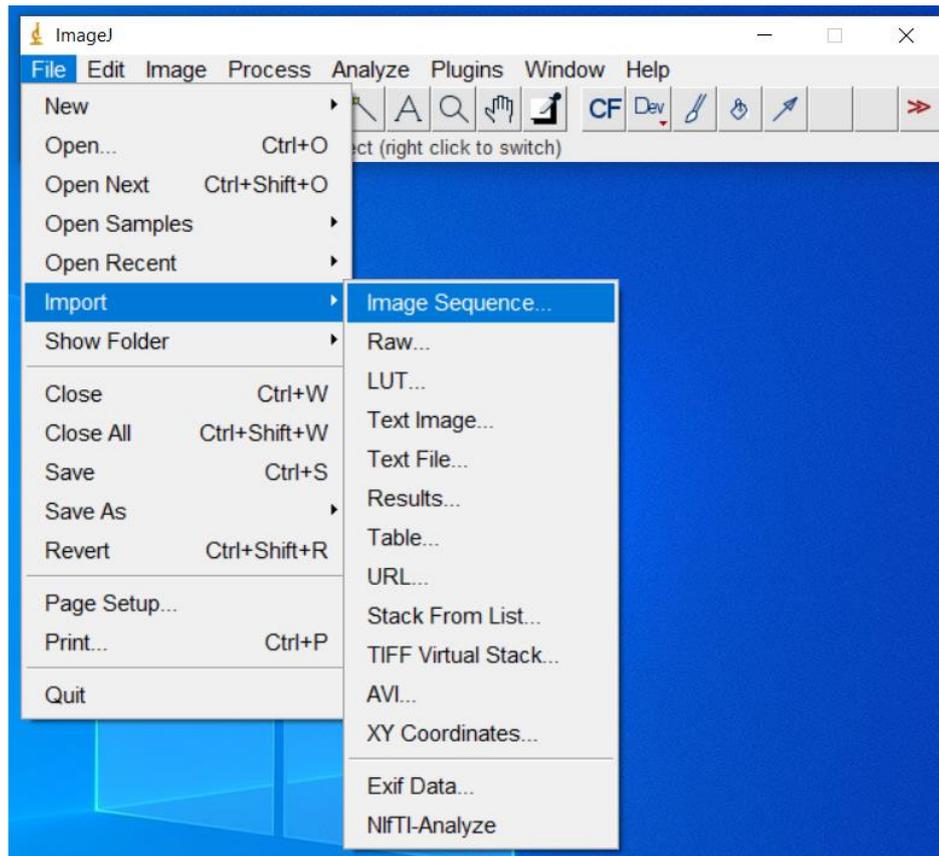


Figura 78: Importando as imagens em sequencia
Fonte: Autoria própria (2021)

Após selecionada a aba *Image Sequence*, o programa abrirá uma janela no qual pode-se informar, através do botão *Browse*, o diretório onde estão armazenadas as imagens para que o *ImageJ* possa iniciar o carregamento das imagens bidimensionais, encerrando assim, o primeiro passo necessário para a reconstrução 3D. Observa-se na figura 79 o caminho do arquivo onde contém as imagens.

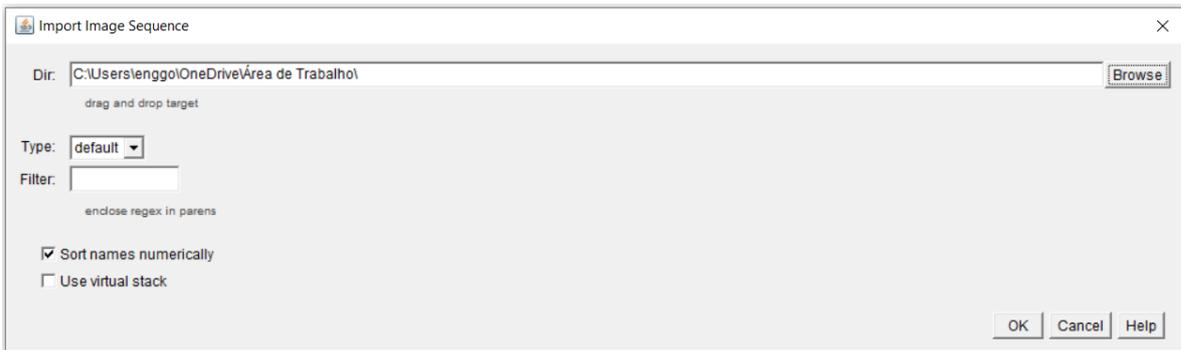


Figura 79: Diretório das imagens
Fonte: Autoria própria (2021)

Após garantir que as imagens foram carregadas corretamente, conforme figura 80, pode-se então iniciar os passos para a reconstrução tridimensional. Note-se que, para o presente trabalho, estão sendo utilizadas 200 imagens para a simulação em questão.

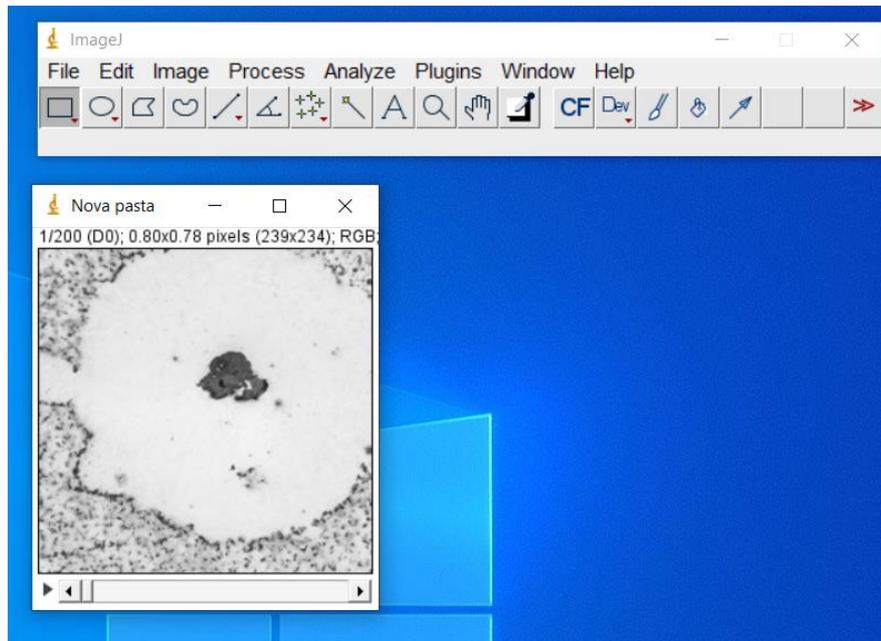


Figura 80: Carregamento das imagens bidimensionais
Fonte: Autoria própria (2021)

Agora, com todas as imagens necessárias para a simulação, pode-se iniciar o processo de reconstrução 3D através do botão *Plugins*, em seguida *3D* e depois a aba *Volume Viewer*. Após seguir essa sequência de passos, o programa iniciará a reconstrução tridimensional a partir das 200 imagens bidimensionais utilizadas. Esses passos podem ser observados na figura 81.

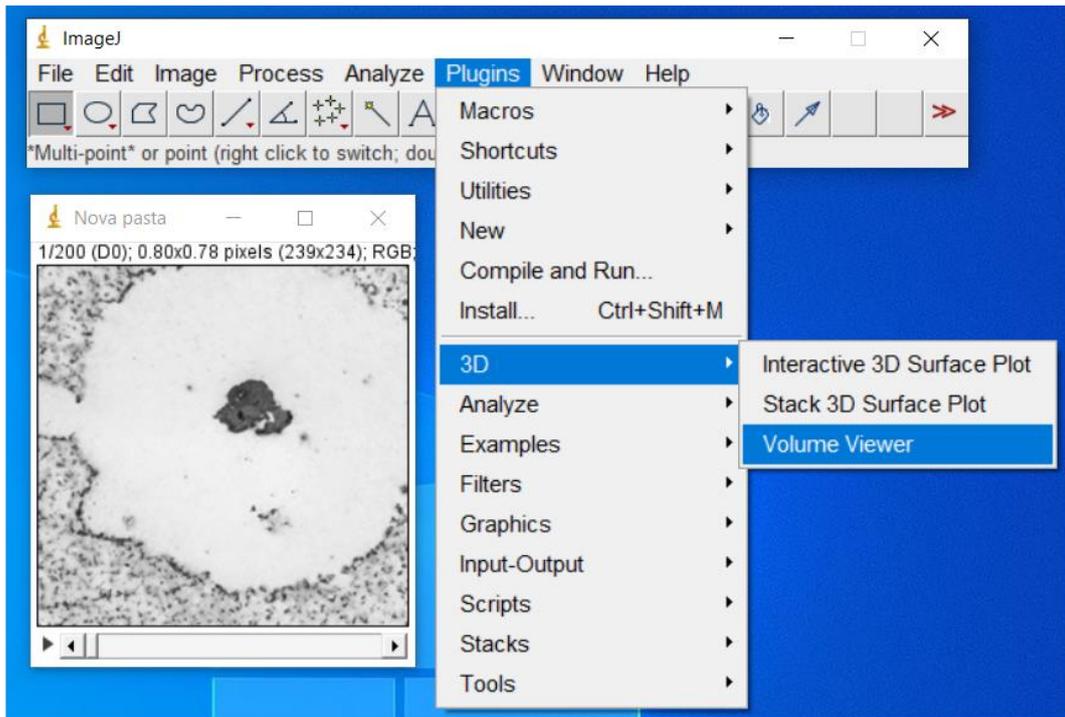


Figura 81: Geração do volume
Fonte: Autoria própria (2021)

Dessa forma, pode-se verificar na figura 82 a reconstrução 3D propriamente dita utilizando as imagens em 2D. Outro ponto a se observar é que o *ImageJ* imprime um volume reconstruído em escalas de cinza.



Figura 82: Volume criado
Fonte: Autoria própria (2021)

Como mostrado na figura 82, o programa gera uma reconstrução tridimensional em escalas de cinzas, porém há a possibilidade de alterarmos as

opções de cores para que facilite as análises metalográficas dependendo do objeto a ser analisado. Observa-se na figura 83 o volume reconstruído com escalas de cores na forma *Thermal LUT*.

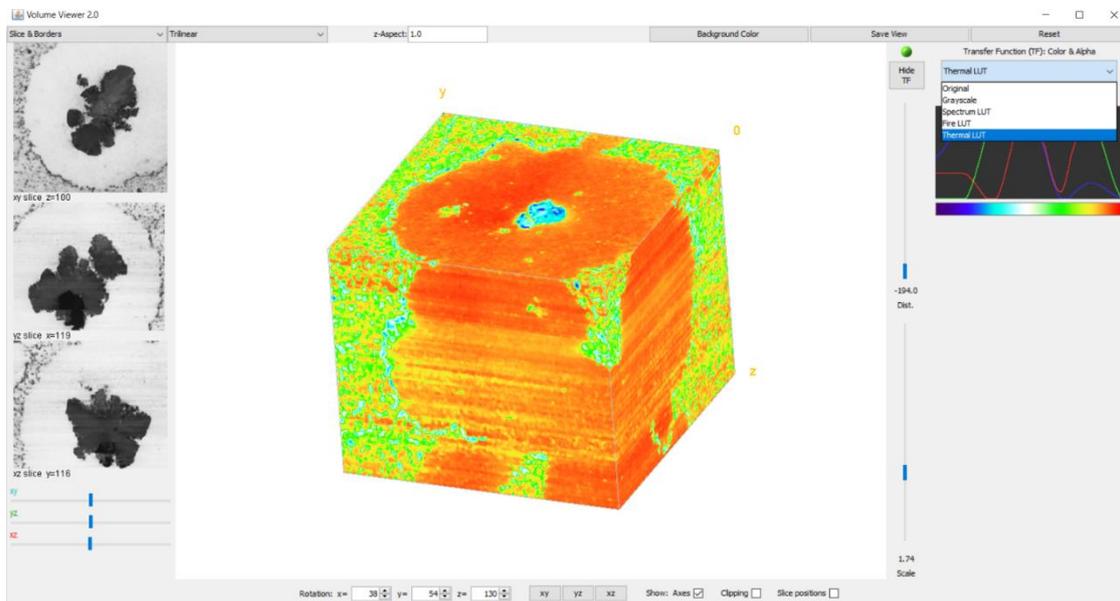


Figura 83: Volume com escala de cores *Thermal LUT*
Fonte: Autoria própria (2021)

Após feita a reconstrução tridimensional, é possível salvar o arquivo gerado através do botão *Save View*. A figura 84 mostra o ícone onde deve-se clicar para que seja possível salvar a imagem tridimensional gerada pela simulação.

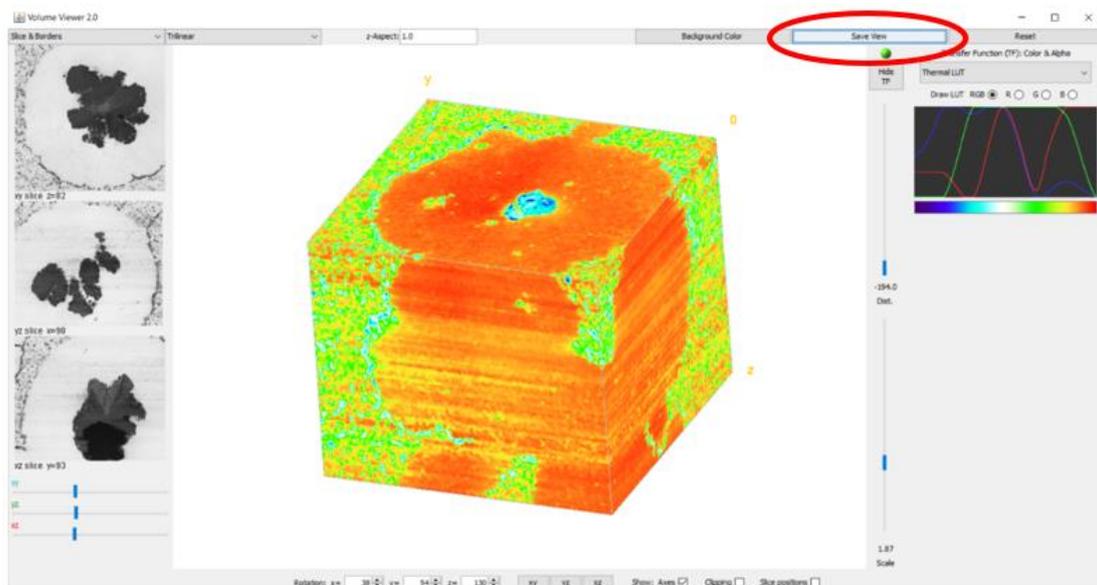


Figura 84: Salvando a imagem tridimensional
Fonte: Autoria própria (2021)

Após clicar no botão *Save View*, o programa irá abrir uma janela com o volume reconstruído. Dessa forma, após esse passo, é possível salvar a imagem tridimensional gerada pelo *ImageJ* em vários tipos de extensão para imagens, como por exemplo, arquivos TIF, JPG, PNG entre outros. Observe na figura 85 as opções de extensão para salvar os arquivos de imagem.

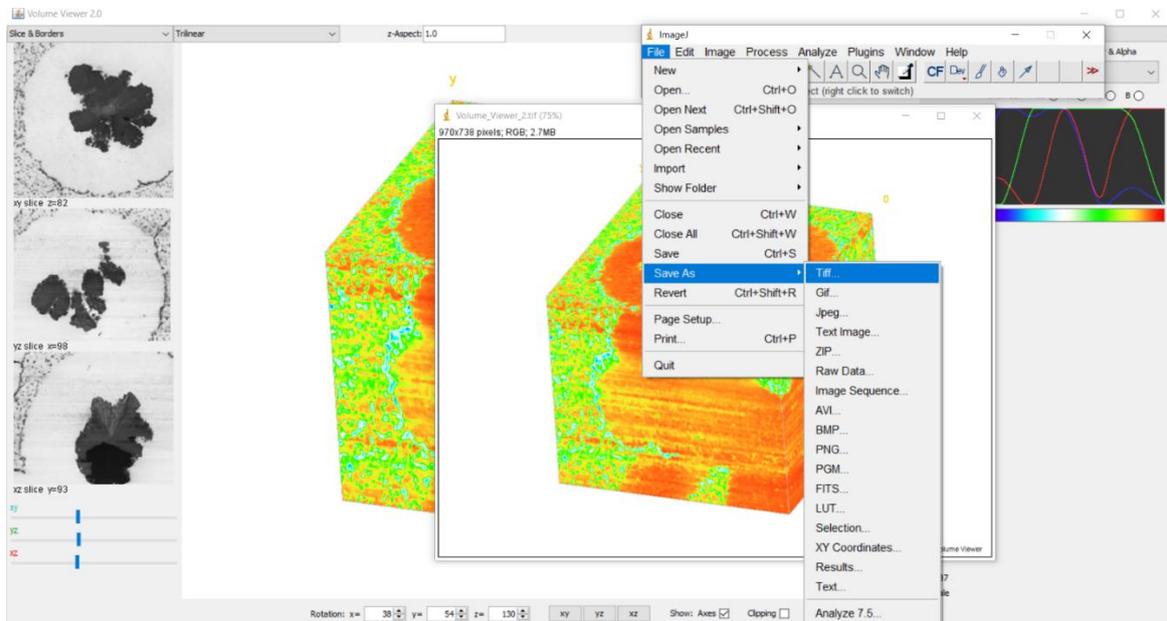
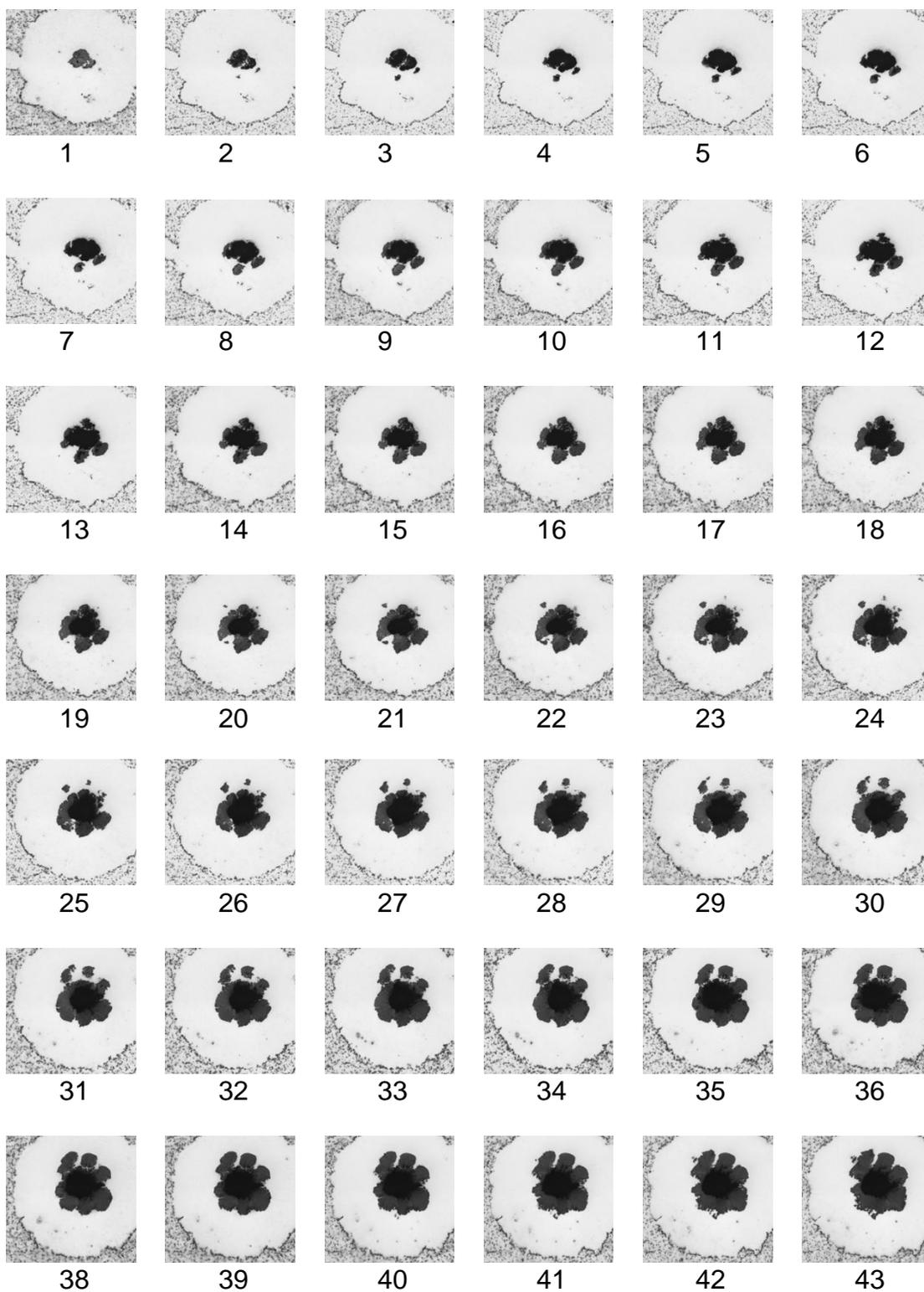
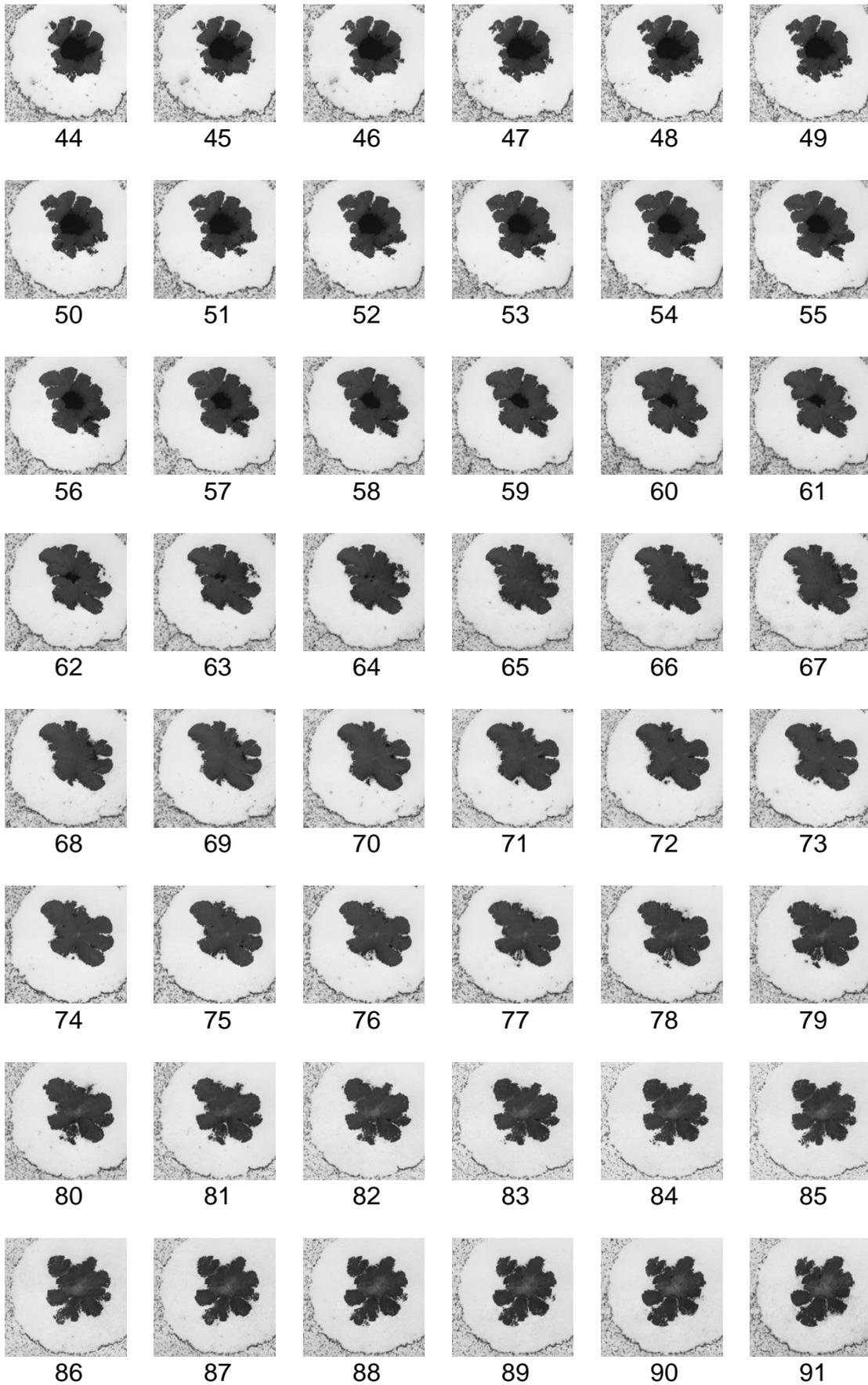
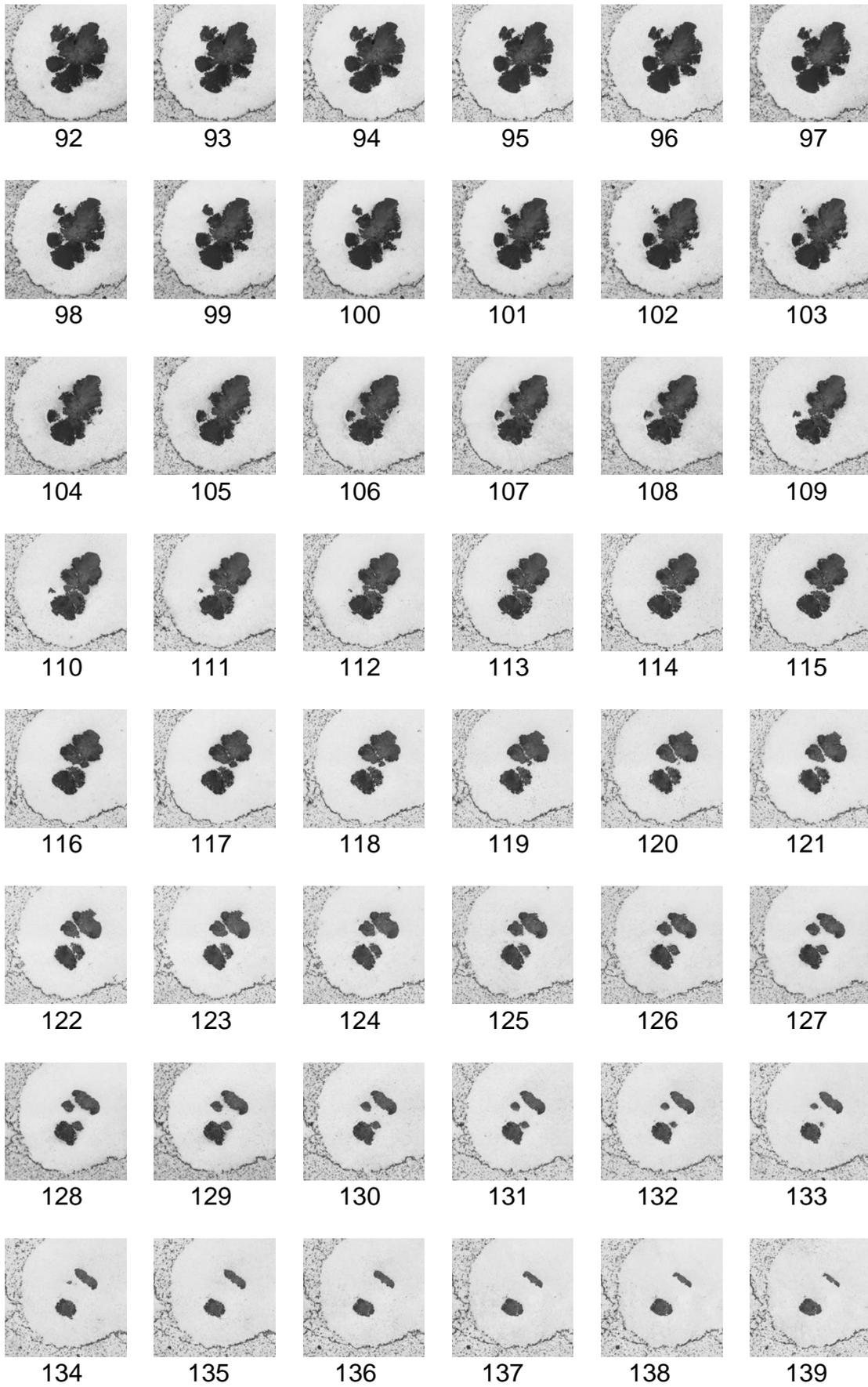


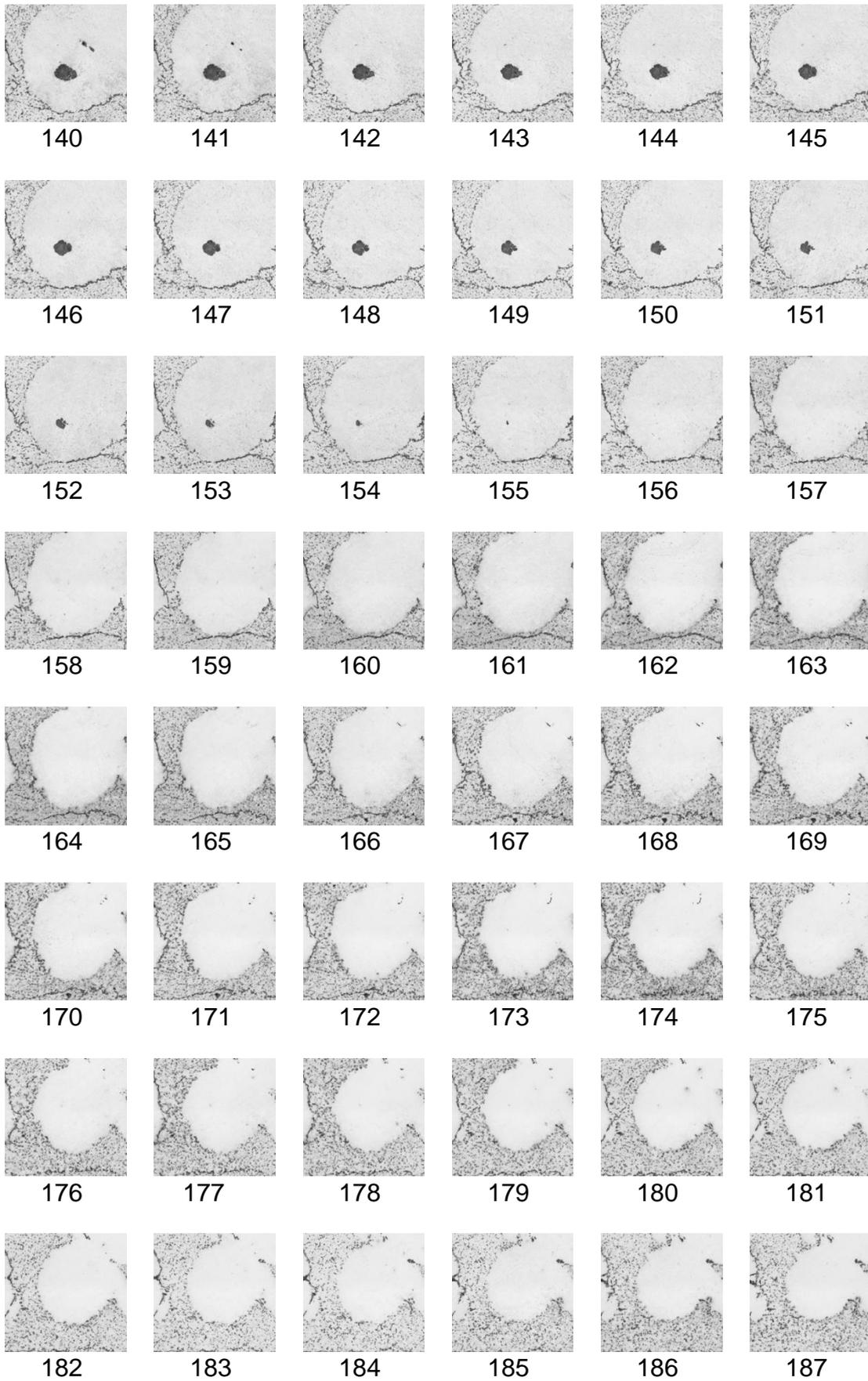
Figura 85: Tipos de extensão para a geração de imagens
Fonte: Autoria própria (2021)

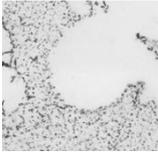
ANEXO A – Imagens utilizadas na Reconstrução 3D - equipamento Robo-Met



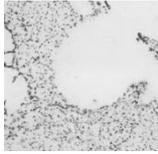




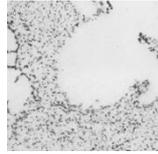




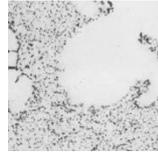
188



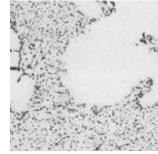
189



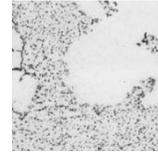
190



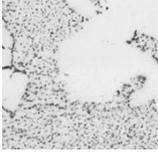
191



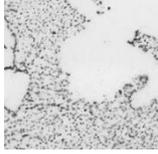
192



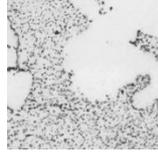
193



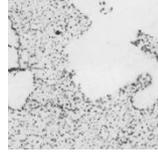
194



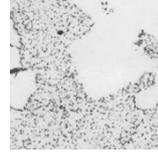
195



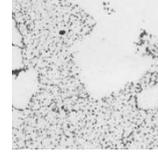
196



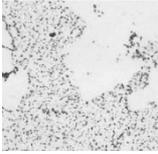
197



198



199



200

ANEXO B – Imagens utilizadas na Reconstrução 3D – equipamento microtomógrafo V-tomex-m

