



**VISÃO COMPUTACIONAL E INTELIGÊNCIA ARTIFICIAL APLICADAS À
AVALIAÇÃO DO PROGRESSO DE CONSTRUÇÕES EM SIMULAÇÕES BIM 4D**

RODRIGO OTÁVIO VALENTE RIBEIRO DA SILVA

DISSERTAÇÃO DE MESTRADO EM ESTRUTURAS E CONSTRUÇÃO CIVIL
DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL

FACULDADE DE TECNOLOGIA

UNIVERSIDADE DE BRASÍLIA

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL

**VISÃO COMPUTACIONAL E INTELIGÊNCIA ARTIFICIAL APLICADAS À
AVALIAÇÃO DO PROGRESSO DE CONSTRUÇÕES EM SIMULAÇÕES
BIM 4D**

RODRIGO OTÁVIO VALENTE RIBEIRO DA SILVA

ORIENTADORA: DR.^a MICHELE TEREZA MARQUES CARVALHO

**DISSERTAÇÃO DE MESTRADO EM ESTRUTURAS E CONSTRUÇÃO
CIVIL**

BRASÍLIA/DF - MARÇO/2023

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL**

**VISÃO COMPUTACIONAL E INTELIGÊNCIA ARTIFICIAL APLICADAS À
AVALIAÇÃO DO PROGRESSO DE CONSTRUÇÕES EM SIMULAÇÕES
BIM 4D**

RODRIGO OTÁVIO VALENTE RIBEIRO DA SILVA

**DISSERTAÇÃO DE MESTRADO SUBMETIDA AO DEPARTAMENTO DE
ENGENHARIA CIVIL E AMBIENTAL DA FACULDADE DE TECNOLOGIA
DA UNIVERSIDADE DE BRASÍLIA COMO PARTE DOS REQUISITOS
NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM
ESTRUTURAS E CONSTRUÇÃO CIVIL.**

APROVADO POR:

Prof.^a Michele Tereza Marques Carvalho, DSc. (ENC-UnB) (Orientadora)

Prof. André Luís Brasil Cavalcante, DSc. (ENC-UnB) (Examinador Interno)

Prof. Kléos Magalhães Lenz César Júnior, DSc. (UFV) (Examinador Externo)

BRASÍLIA/DF - MARÇO/2023

FICHA CATALOGRÁFICA

DA SILVA, RODRIGO OTÁVIO VALENTE RIBEIRO

Visão computacional e inteligência artificial aplicadas à avaliação do progresso de construções em simulações BIM 4D. [Distrito Federal] 2023.

xx, 148 p., 210 x 297 mm (ENC/FT/UnB, Mestre em Estruturas e Construção Civil, 2022).

Dissertação de Mestrado – Universidade de Brasília. Faculdade de Tecnologia. Departamento de Engenharia Civil e Ambiental.

- | | |
|---|----------------------------|
| 1. Building Information Modelling (BIM) | 2. Inteligência artificial |
| 3. Monitoramento de construções | 4. Redes Neurais |
| 5. Redes Neurais Convolucionais | 6. Visão computacional |

I. ENC/FT/UnB

II. Título (Mestre)

REFERÊNCIA BIBLIOGRÁFICA

DA SILVA, R. O. V. R. (2023). Visão computacional e inteligência artificial aplicadas à avaliação do progresso de construções em simulações BIM 4D. Dissertação de Mestrado em Estruturas e Construção Civil, Publicação E.DM – 03A/23, Departamento de Engenharia Civil e Ambiental, Universidade de Brasília, Brasília, DF, 148 p.

CESSÃO DE DIREITOS

AUTOR: Rodrigo Otávio Valente Ribeiro da Silva

TÍTULO: Visão computacional e inteligência artificial aplicadas à avaliação do progresso de construções em simulações BIM 4D

GRAU: Mestre

ANO: 2023

É concedida à Universidade de Brasília a permissão para reproduzir cópias desta dissertação de mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. Os autores reservam outros direitos de publicação e nenhuma parte desta dissertação de mestrado pode ser reproduzida sem a autorização por escrito do autor.

Rodrigo Otávio Valente Ribeiro da Silva

Brasília/DF – Brasil

AGRADECIMENTOS

Neste momento de grande alegria, em que finalizo uma importante etapa de minha jornada acadêmica, gostaria de agradecer a algumas pessoas que tornaram a caminhada mais fácil, prazerosa e divertida.

Primeiramente, gostaria de agradecer à minha orientadora, professora Michele, pela orientação ao longo do desenvolvimento deste trabalho. As discussões sobre o tema, os conselhos e as ideias foram de fundamental importância para o amadurecimento desta pesquisa. Agradeço também pela empatia, compreensão e paciência, que foram importantes para que, junto com a dissertação, eu pudesse encarar também outros desafios em minha vida.

Agradeço também ao Fábio, colega de minha orientadora, que disponibilizou arquivos, informações e dados primordiais para a condução desta pesquisa. Pela atenção e disponibilidade, o meu muito obrigado.

Gostaria igualmente de agradecer aos meus amigos de mestrado por me acompanharem nessa trajetória. Me ajudaram no estudo para provas, em trabalhos, no desenvolvimento de artigos, em dúvidas sobre o dia a dia do mestrado. Vocês tornaram a jornada mais divertida e alegre, e desejo um sucesso gigantesco para cada um de vocês.

Agradeço também à minha família por todo o apoio, incentivo, e todas as conversas. Minhas conquistas são conquistas de todos nós. Meu pai, minha mãe, minha irmã e meu irmão. Amo muito todos vocês.

O meu muito obrigado também aos professores e funcionários do Programa de Pós-Graduação em Estruturas e Construção Civil (PECC), por todo o auxílio ao longo do meu mestrado.

O meu carinho também à Universidade de Brasília, minha casa já há muitos anos, e um lugar que eu tanto amo.

O meu agradecimento também a Deus, por me iluminar e guiar o meu caminho, me dando todas as oportunidades e condições para lutar diariamente pelos meus sonhos, alguns já se realizando!

Há alguns anos atrás, terminei os agradecimentos do meu Trabalho de Conclusão de Curso em engenharia civil na UnB com a seguinte frase: “Vamos à próxima etapa”. Confesso que, dessa vez, queimei a largada, e já estou trabalhando em meu próximo projeto. Mas para não perder o costume...

Vamos à próxima etapa.

RESUMO

A indústria da construção civil é notadamente conhecida pelo seu baixo grau de inovação tecnológica. Entretanto, essa realidade tem se alterado com o uso do Building Information Modelling, da inteligência artificial e da visão computacional em diversos campos dessa indústria, sendo que essas tecnologias estão sendo utilizadas tanto de forma independente como integradas. Nesse contexto, tem-se que o monitoramento e acompanhamento de construções é uma área que está se beneficiando constantemente dessas mudanças. O controle de obras, nesse sentido, se trata de uma área de fundamental importância para o êxito de um projeto, uma vez que permite acompanhar se o que está sendo executado em campo está compatível com o que foi planejado, de forma a identificar desvios e tomar decisões acertadas para mitigar atrasos. A visão computacional, o BIM 4D, o aprendizado de máquina e a Internet das Coisas são exemplos de tecnologias que estão sendo utilizadas para auxiliar os gestores de obras nessa etapa do processo construtivo. Entretanto, tem-se que, para que esses recursos sejam utilizados de forma adequada no acompanhamento de obras, são necessários estudos e pesquisas sobre cada um deles, bem como sobre sua integração. Dessa forma, essa dissertação propõe um uso da visão computacional e da inteligência artificial aplicadas à avaliação do progresso de obras em uma simulação BIM 4D. Para tanto, foi criado um banco de dados de imagens de simulações de uma construção, cada uma com o respectivo progresso da construção. Em seguida, essas imagens foram utilizadas para treinar 3 redes neurais convolucionais com o objetivo de obter o progresso da construção em cada imagem. Os modelos de aprendizado de máquina treinados foram então avaliados estatisticamente, de forma que foi possível analisar se o algoritmo foi capaz de identificar as principais características das imagens, além de avaliar a ocorrência de *underfitting* e de *overfitting* nos modelos. Posteriormente, foi feita uma análise dos *feature maps* gerados pelas redes neurais convolucionais, de forma a buscar compreender melhor os resultados do modelo. Como contribuição científica, essa dissertação explora a integração entre o BIM e as redes neurais convolucionais, mostrando que estas podem ser treinadas com imagens de modelos BIM e fornecer resultados precisos. Além disso, este estudo utiliza as redes neurais convolucionais para problemas de regressão, algo pouco explorado na literatura científica.

Palavras-chave: BIM; Controle de Construções; Inteligência Artificial; Inteligência Artificial Explicável; Redes Neurais; Redes Neurais Convolucionais; Visão Computacional.

ABSTRACT

The construction industry is notoriously known for its low degree of technological innovation. However, this reality has changed with the use of Building Information Modelling, artificial intelligence and computer vision in several fields of this industry, and these technologies are being used both independently and integrated. In this context, construction monitoring is one area that is constantly benefiting from these changes. Construction control, in this sense, is an area of fundamental importance for the success of a project, since it allows the monitoring of whether what is being executed in the field is compatible with what was planned, in order to identify deviations and make the right decisions to mitigate delays. Computer vision, 4D BIM, machine learning and the Internet of Things are examples of technologies that are being used to help construction managers at this stage of the building process. However, for these resources to be used properly in construction monitoring, studies and research are needed on each of them, as well as on their integration. Thus, this dissertation proposes the use of computer vision and artificial intelligence applied to the evaluation of construction progress in a 4D BIM simulation. To this end, a database of images of simulations of a construction was created, each with its respective construction progress. Then, these images were used to train 3 convolutional neural networks with the objective of obtaining the construction progress in each image. The trained machine learning models were then statistically evaluated, so that it was possible to analyze whether the algorithm was able to identify the main characteristics of the images, and analyze the occurrence of underfitting and overfitting in the models. Subsequently, an analysis of the feature maps generated by the convolutional neural networks was performed in order to better understand the model results. As a scientific contribution, this dissertation explores the integration between BIM and convolutional neural networks, showing that these networks can be trained with BIM model images and provide accurate results. In addition, this study uses convolutional neural networks for regression problems, something little explored in the scientific literature.

Keywords: BIM; Building Control; Artificial Intelligence; Explainable Artificial Intelligence; Neural Networks; Convolutional Neural Networks; Computer Vision.

LISTA DE FIGURAS

Figura 1: Elementos do processo BIM.....	27
Figura 2: Áreas e subdivisões da inteligência artificial	32
Figura 3: Exemplo de neurônio artificial	33
Figura 4: Exemplo de rede neural.....	34
Figura 5: Exemplo de rede neural convolucional	35
Figura 6: Modelo estrutural inserido no software Autodesk Revit.....	46
Figura 7: Fôrma de pilar utilizada no projeto.....	48
Figura 8: Família de bandeja utilizada no projeto	48
Figura 9: Comando Append no Autodesk Navisworks	49
Figura 10: Lista de formatos suportados pelo Autodesk Navisworks	49
Figura 11: Detalhamento das modificações realizadas no cronograma da construção...	50
Figura 12: Ferramenta TimeLiner do Autodesk Navisworks.....	50
Figura 13: Cronograma inserido no Autodesk Navisworks	51
Figura 14: Projeto inserido no Autodesk Navisworks, destacando o grupo de fôrmas de pilares do terceiro pavimento	52
Figura 15: Primeira etapa metodológica da pesquisa	53
Figura 16: Segunda etapa metodológica da pesquisa	58
Figura 17: Terceira etapa metodológica da pesquisa.....	60
Figura 18: Quarta etapa metodológica da pesquisa	61
Figura 19: Esquema das etapas metodológicas da pesquisa.....	63
Figura 20: Modelo tridimensional estrutural com elementos temporários da construção	64
Figura 21: Modelo BIM no software Auodesk Navisworks.	65
Figura 22: Exemplo de simulação gerada no Autodesk Navisworks	65
Figura 23: Exemplo de imagem gerada na simulação	66
Figura 24: Exemplo de simulação com imagens sequenciais da construção	67
Figura 25: Exemplo de simulação BIM 4D.....	72
Figura 26: Primeira imagem analisada.....	78
Figura 27: Feature maps da primeira imagem analisada após a primeira camada de convolução.....	78
Figura 28: Feature maps da primeira imagem analisada após a primeira camada de pooling	78

Figura 29: Feature maps da primeira imagem analisada após a segunda camada de convolução.....	78
Figura 30: Segunda imagem analisada.....	79
Figura 31: Feature maps da segunda imagem analisada após a primeira camada de convolução.....	80
Figura 32: Feature maps da segunda imagem analisada após a primeira camada de pooling	80
Figura 33: Feature maps da segunda imagem analisada após a segunda camada de convolução.....	80
Figura 34: Terceira imagem analisada.....	80
Figura 35: Feature maps da terceira imagem analisada após a primeira camada de convolução.....	81
Figura 36: Feature maps da terceira imagem analisada após a primeira camada de pooling	81
Figura 37: Feature maps da terceira imagem analisada após a segunda camada de convolução.....	81
Figura 38: Simulação 1	97
Figura 39: Simulação 2	98
Figura 40: Simulação 3	99
Figura 41: Simulação 4	100
Figura 42: Simulação 5	101
Figura 43: Simulação 6	102
Figura 44: Simulação 7	103
Figura 45: Simulação 8	104
Figura 46: Simulação 9	105
Figura 47: Simulação 10	106
Figura 48: Simulação 11	107
Figura 49: Simulação 12	108
Figura 50: Simulação 13	109
Figura 51: Simulação 14	110
Figura 52: Simulação 15	111
Figura 53: Simulação 16	112
Figura 54: Simulação 17	113
Figura 55: Simulação 18	114

Figura 56: Simulação 19	115
Figura 57: Simulação 20	116
Figura 58: Simulação 21	117
Figura 59: Simulação 22	118
Figura 60: Simulação 23	119
Figura 61: Simulação 24	120
Figura 62: Simulação 25	121
Figura 63: Simulação 26	122
Figura 64: Simulação 27	123
Figura 65: Simulação 28	124
Figura 66: Simulação 29	125
Figura 67: Simulação 30	126
Figura 68: Simulação 31	127
Figura 69: Simulação 32	128
Figura 70: Simulação 33	129
Figura 71: Simulação 34	130
Figura 72: Simulação 35	131
Figura 73: Simulação 36	132
Figura 74: Simulação 37	133
Figura 75: Simulação 38	134
Figura 76: Simulação 39	135
Figura 77: Simulação 40	136
Figura 78: Estrutura da Rede Neural Convolutacional 1 (RNC_1).....	142
Figura 79: Estrutura da Rede Neural Convolutacional 2 (RNC_2).....	143
Figura 80: Estrutura da Rede Neural Convolutacional 3 (RNC_3).....	144

LISTA DE TABELAS

Tabela 1: Estrutura da rede neural convolucional LeNet.....	54
Tabela 2: Ferramentas de programação utilizadas nessa pesquisa	56
Tabela 3: Ferramentas utilizadas em cada etapa da pesquisa.....	57
Tabela 4: Avaliação do modelo de aprendizado de máquina.....	60
Tabela 5: Etapas da pesquisa.....	62
Tabela 6: Progresso calculado para cada imagem de cada simulação	67
Tabela 7: Estrutura das redes neurais convolucionais	70
Tabela 8: Quantidade de parâmetros treináveis em cada RNC treinada.....	70
Tabela 9: Comparação do progresso real e estimado para a simulação exemplo.....	73
Tabela 10: Avaliação estatística das redes neurais convolucionais.....	74

SIGLAS E ABREVIACOES

3D	Três Dimenses
4D	Quatro Dimenses
5D	Cinco Dimenses
6D	Seis Dimenses
7D	Sete Dimenses
8D	Oito Dimenses
9D	Nove Dimenses
10D	Dez Dimenses
BIM	Building Information Modelling
IA	Inteligncia Artificial
IFC	Industry Foundation Classes
MAE	<i>Mean Absolute Error</i>
RNA	Rede Neural Artificial
RNC	Rede Neural Convolutacional
RMSE	<i>Root Mean Squared Error</i>
xAI	Inteligncia Artificial Explicvel

SUMÁRIO

1	INTRODUÇÃO	15
1.1	CONTEXTUALIZAÇÃO	15
1.2	OBJETIVO GERAL	17
1.3	OBJETIVOS ESPECÍFICOS	18
1.4	ORIGINALIDADE	18
1.5	RESTRIÇÕES DA PESQUISA	19
1.6	ESTRUTURA DO TRABALHO	20
2	REVISÃO DA LITERATURA	22
2.1	CONTROLE E MONITORAMENTO DE CONSTRUÇÕES	22
2.1.1	Visão computacional aplicada ao monitoramento de construções	25
2.2	BUILDING INFORMATION MODELLING	26
2.2.1	Dimensões do BIM	28
2.2.2	BIM no controle de construções	30
2.3	INTELIGÊNCIA ARTIFICIAL	31
2.3.1	Redes Neurais Artificiais	32
2.3.2	Redes Neurais Convolucionais	35
2.3.3	Redes Neurais Artificiais na construção civil	36
2.4	BIM E IA	38
2.5	Síntese da Revisão da Literatura	42
3	METODOLOGIA	44
3.1	Parte 1 – Desenvolvimento de um banco de dados de imagens de uma simulação BIM 4D com o respectivo progresso da construção	46
3.2	Parte 2 – Desenvolvimento da estrutura e treinamento das redes neurais convolucionais	53
3.3	Parte 3 - Avaliação estatística das redes neurais convolucionais geradas	58

3.4	Parte 4 - Análise dos <i>feature maps</i> gerados pelas redes neurais convolucionais	61
3.5	Visão geral do desenvolvimento do trabalho	62
4	RESULTADOS E DISCUSSÃO	64
4.1	Parte 1 – Desenvolvimento de um banco de dados de imagens de uma simulação BIM 4D com o respectivo progresso da construção	64
4.2	Parte 2 – Desenvolvimento da estrutura e treinamento das redes neurais convolucionais	68
4.3	Parte 3 - Avaliação estatística das redes neurais convolucionais geradas	71
4.4	Parte 4 - Interpretação dos resultados das redes neurais convolucionais.	76
5	CONCLUSÕES	82
6	TRABALHOS FUTUROS	85
	REFERÊNCIAS BIBLIOGRÁFICAS	87
	APÊNDICE I	97
	APÊNDICE II	137
	APÊNDICE III	140
	APÊNDICE IV	142
	APÊNDICE V	145
	APÊNDICE VI	148

1 INTRODUÇÃO

Essa dissertação de mestrado foi desenvolvida no Programa de Pós-Graduação em Estruturas e Construção Civil (PECC) da Universidade de Brasília (UnB) sob a orientação da Professora Dra. Michele Tereza Marques Carvalho, se tratando de uma pesquisa na linha de gestão e sustentabilidade na construção civil.

1.1 CONTEXTUALIZAÇÃO

O controle do progresso de uma construção é uma tarefa de fundamental importância no transcurso de uma obra. Esse monitoramento consiste não só em verificar se as diversas atividades previstas no cronograma da construção foram executadas, mas também se foram realizadas no prazo correto, com os custos previstos e na qualidade adequada. O objetivo de se obter informações da obra por meio desse controle é auxiliar os agentes envolvidos no projeto em sua tomada de decisões, de forma que sejam acertadas e embasadas.

Uma definição para monitoramento do progresso de obras, estabelecido por Del Pico (2013), é que essa atividade se trata de um conjunto de etapas com o objetivo de se determinar a evolução das atividades a serem executadas na obra, comparadas com o que foi previamente planejado. Entretanto, uma vez que esse controle é feito de forma visual, com alta dependência da experiência e capacitação do engenheiro que o está realizando, ele pode muitas vezes não corresponder com exatidão ao verdadeiro progresso da construção (NAVON; SACKS, 2007). Além disso, esse método tradicional para o controle de obras, além de muitas vezes não fornecer um resultado preciso, se trata de um processo caro, demorado, desorganizado e de baixa qualidade (ARIF; KHAN, 2020).

O desafio de se monitorar construções possui reflexos tanto em obras públicas quanto privadas. No âmbito público, a necessidade de métodos mais eficientes de fiscalização pode ser exemplificada por um relatório divulgado pelo Tribunal de Contas da União (TCU) no ano de 2014. Neste relatório, o TCU divulgou que, de um total de 668 auditorias em obras públicas realizadas entre 2011 e 2014, 48 delas (ou seja, 7,2%) possuíam atrasos injustificáveis em suas obras e serviços, 42 (6%) possuíam execução de serviços com qualidade deficiente, 253 (38%) apresentavam sobrepreço ou estavam superfaturadas e 77 (12%) apresentavam fiscalização deficiente ou omissa (MATOS, 2016).

Diversas alternativas estão sendo estudadas e aplicadas para tornar o controle de obras uma tarefa mais econômica e rápida, entre elas o uso de tecnologias e processos como o Building Information Modelling (BIM), a realidade aumentada (RA), o escaneamento a laser, o uso de veículos aéreos não-tripulados (VANTs), a inteligência artificial (IA) e a visão computacional. Além disso, tem-se que, muitas vezes, essas tecnologias e processos são utilizados de forma integrada, uma aumentando o potencial da outra.

Mostafa e Hagazy (2021), por exemplo, apresentaram um Mapeamento Sistemático da Literatura (MSL) para identificar o estado-da-arte da aplicação da visão computacional na construção civil. Nessa pesquisa, foi possível concluir que essa tecnologia foi utilizada majoritariamente nos campos de segurança na construção, monitoramento do progresso de obras e avaliação de danos. A técnica mais utilizada de visão computacional identificada pelos pesquisadores foram as redes neurais convolucionais (RNCs), uma técnica de inteligência artificial para análise de imagens.

O BIM também vem sendo amplamente utilizado no campo do controle de construções. Com o uso de softwares BIM, como por exemplo o Autodesk Navisworks, é possível simular computacionalmente as etapas de uma obra e visualizar por meio de vídeos ou de uma sequência de imagens o progresso de uma construção, integrando o modelo tridimensional com o planejamento da obra, de forma a indicar a respectiva data de cada atividade no cronograma e inclusive com a possibilidade de se inserir informações sobre os custos envolvidos. Conforme a obra for sendo executada, é possível inserir na ferramenta BIM a data em que cada tarefa foi efetivamente realizada, permitindo uma comparação entre o que foi planejado e o que foi executado. Dessa forma, tem-se uma forma intuitiva e visual de se monitorar a construção, facilitando a identificação de atrasos na execução do projeto (MAINI, 2023).

Nesse contexto, também há exemplos de integração do BIM com a visão computacional para o controle de construções. Em um estudo conduzido por Wei *et al.* (2022), por exemplo, utilizou-se uma técnica de inteligência artificial, do campo da visão computacional, para segmentação de imagens e avaliar de forma automática o progresso da construção de paredes, sendo que esses resultados eram posteriormente transferidos para um *software* BIM.

Nessa conjuntura de integração de tecnologias e processos, uma revisão da literatura conduzida por da Silva e Carvalho (2022) revelou que o BIM e as redes neurais convolucionais (técnica específica de visão computacional e inteligência artificial) são

uma combinação eficiente para solução de diversos problemas na construção civil, especialmente na área de controle de construções e classificação de elementos em modelos BIM. Apesar disso, foi revelado na pesquisa que ainda são poucos os estudos que efetivamente associam essas duas tecnologias, de forma que há uma lacuna no conhecimento entre essas áreas.

Também vale destacar que, apesar de ser possível a integração e o uso conjunto de diversas tecnologias para o controle de obras, essa associação não é perfeita, de forma que há a perda de informações e atrasos quando há a transferência de dados de uma plataforma para outra. Esses problemas podem decorrer tanto da grande quantidade de etapas para a integração entre as tecnologias, como pela falta de interoperabilidade entre elas.

A falha na associação entre tecnologias no âmbito de controle de construções pode ser exemplificada pelo monitoramento da construção de uma parede. Para que seja feito esse controle, tem-se que, após a construção do elemento no canteiro de obras, alguma tecnologia deve ser utilizada para capturar a construção dessa parede e transformar essa informação em um dado computacional. Posteriormente, esse dado deve ser associado a uma tarefa do cronograma da construção em um *software* de planejamento e monitoramento de obras e, em seguida, o cronograma deve ser atualizado conforme a construção desse elemento. Nesse contexto, pode haver falhas na captura da informação da construção da parede, na associação do objeto construído com a respectiva tarefa do cronograma, bem como na identificação de qual o real progresso na execução dessa tarefa.

Dessa forma, pode-se verificar que é de fundamental importância o estudo da associação entre as diversas tecnologias quando se busca integrá-las para se solucionar um problema da construção civil. No caso específico do controle e monitoramento de uma construção, essa lacuna é destacada no estudo da integração entre o BIM e as redes neurais convolucionais. Dessa forma, essa pesquisa pretende investigar a associação entre essas duas tecnologias, em um contexto de monitoramento de construções, uma vez que são escassas as pesquisas que avaliam essas duas técnicas utilizadas em conjunto.

1.2 OBJETIVO GERAL

O objetivo geral deste trabalho é avaliar a integração entre o BIM e as redes neurais convolucionais para regressão em um contexto de monitoramento e controle de obras, com o uso de imagens resultantes de simulações computacionais de uma

construção. A obra a ser simulada digitalmente no *software* BIM se trata do projeto estrutural de um edifício residencial de 6 pavimentos localizado no Distrito Federal.

1.3 OBJETIVOS ESPECÍFICOS

São objetivos específicos dessa pesquisa:

- Criar um banco de dados de imagens com as etapas da construção, geradas pela simulação BIM 4D, associadas ao respectivo percentual de progresso da obra em cada imagem proveniente da simulação;
- Avaliar a performance das redes neurais convolucionais treinadas com imagens da simulação computacional da construção, verificando a precisão do modelo, bem como a ocorrência de *overfitting* e de *underfitting*;
- Examinar a aplicabilidade das redes neurais convolucionais para problemas de regressão na área da construção civil, especificamente previsão do progresso de construções em simulações BIM 4D;
- Avaliar os resultados das redes neurais convolucionais por meio de mapas de características (*feature maps*), buscando compreender os resultados gerados pelos modelos de inteligência artificial.

1.4 ORIGINALIDADE

Existem diversas pesquisas que utilizam as redes neurais convolucionais. Entretanto, tem-se que a maioria dessas pesquisas utiliza essa técnica com o objetivo de realizar classificações, como por exemplo, identificar números escritos à mão (CIREGAN *et al.*, 2012), reconhecimento de expressões faciais (MATSUGU *et al.*, 2003), reconhecimento facial humano (LAWRENCE *et al.*, 1997), classificar imagens entre diferentes tipos de animais, objetos e alimentos (KRIZHEVSKY *et al.*, 2017), entre outras aplicações.

Todavia, tem-se que as redes neurais convolucionais, por sua vez, também podem ser utilizadas para se fazer regressões, ou seja, a partir de uma imagem, estabelecer relações entre os elementos identificados na imagem e, assim, estimar valores contínuos. Apesar dessa possibilidade, tem-se que o uso de redes neurais convolucionais para

regressão ainda é muito pouco utilizado em pesquisas em geral, seja em qualquer campo de estudo.

Além do exposto, tem-se que a maior parte das pesquisas conduzidas e que propõem uma integração entre o BIM e a visão computacional, normalmente capturam informações do ambiente para posteriormente inseri-las dentro de uma plataforma BIM. Entretanto, nessa pesquisa, tem-se que o algoritmo de visão computacional está extraindo e sendo treinado com informações visuais do modelo BIM. Dessa forma, tem-se que o modelo BIM está sendo utilizado neste estudo como um banco de dados primário, de onde informações são extraídas, e não como um repositório de informações, onde são inseridas.

Dessa maneira, a originalidade desta pesquisa consiste no uso das redes neurais convolucionais para se fazer regressões matemáticas no campo da construção civil, além de integrar essa técnica com simulações BIM 4D. Dessa forma, além de explorar a integração entre o BIM e as RNCs, este estudo também tem como objetivo avançar na compreensão sobre o uso das redes neurais convolucionais quando utilizadas para regressões.

1.5 RESTRIÇÕES DA PESQUISA

Uma das etapas metodológicas desta pesquisa é o desenvolvimento de um modelo tridimensional da construção. Nesse sentido, a principal limitação se trata dos componentes computacionais tridimensionais necessários para a modelagem do canteiro de obras em um software BIM. Para o desenvolvimento deste estudo, foi obtido um *template* do software Autodesk Revit específico para a modelagem dos elementos temporários da construção, como por exemplo fôrmas de pilares, vigas, escoramentos, bandejas, entre outros. Entretanto, em que pese a existência desses componentes no *template*, tem-se que há diferenças significativas entre os objetos contidos nesse arquivo padrão e os que efetivamente foram utilizados durante a obra, como foi possível verificar em fotografias do canteiro, com divergências nos materiais utilizados, na cor dos objetos, tamanho, entre outros. Dessa forma, em que pese seja possível fazer uma modelagem do canteiro de obras que se aproxime do canteiro efetivamente construído, não será possível fazer uma reprodução exata da obra real no modelo digital tridimensional.

Outra limitação com relação à modelagem do canteiro se trata do fato de que não foi fornecido o projeto de alguns elementos do canteiro de obras, como por exemplo

escoramentos, linha de vida, bandejas, entre outros, o que poderá resultar em uma simulação da construção divergente do que ocorreu na obra efetivamente.

Também houve limitações quanto ao tamanho do arquivo tridimensional que será produzido, uma vez que muitos elementos deverão ser modelados, além do próprio projeto estrutural do edifício, o que dificultaria a manipulação do arquivo tridimensional e sua transposição para outros softwares BIM. Dessa forma, tem-se que para efetividade da pesquisa e limitações computacionais para manipulação de arquivos BIM com muitas informações, não serão modelados todos os elementos do canteiro de obras para a simulação da construção.

Por fim, tem-se que esta pesquisa foi realizada utilizando-se somente o projeto estrutural do edifício fornecido, de forma que a simulação da construção não envolve o projeto arquitetônico, tampouco o de instalações prediais.

1.6 ESTRUTURA DO TRABALHO

Este trabalho está estruturado da seguinte forma:

Capítulo 1 - Introdução: contextualização geral do assunto, com a posterior apresentação dos objetivos gerais e específicos da pesquisa, a originalidade deste trabalho e suas limitações.

Capítulo 2 - Revisão da literatura: análise aprofundada dos principais conceitos, temáticas e pesquisas abordadas neste trabalho, como Building Information Modelling, inteligência artificial, redes neurais artificiais, redes neurais convolucionais, bem como o uso, aplicação e integração entre essas ferramentas para o controle e monitoramento do progresso de construções.

Capítulo 3 - Metodologia: apresentação de forma textual e por meio de fluxogramas de como a pesquisa será conduzida e como os resultados obtidos serão analisados. Detalhará as fases do projeto e sua execução, que serão a criação do banco de dados de imagens da simulação BIM 4D e seu respectivo progresso; desenvolvimento e treinamento das redes neurais convolucionais; avaliação estatística dos modelos de aprendizado de máquina; e interpretação dos resultados das redes neurais convolucionais.

Capítulo 4 – Resultados e discussão: apresentação dos resultados do desenvolvimento da pesquisa, com apresentação detalhada do banco de dados gerado, estrutura das redes neurais convolucionais treinadas, resultados da análise estatística e

interpretação dos resultados dos modelos por meio dos *feature maps* gerados pelas redes neurais convolucionais.

Capítulo 5 – Conclusão: exposição dos resultados da pesquisa de forma consolidada, com discussão sobre seus principais aspectos, dificuldades, bem como avaliação do progresso científico que o estudo trouxe para a área da construção civil.

Capítulo 6 – Trabalhos futuros: apresentação de futuras pesquisas que podem ser desenvolvidas com os resultados deste trabalho, seja por meio de uma continuação direta deste estudo ou a aplicação dos resultados obtidos em outros projetos com temáticas distintas.

Apêndices – Apresentação de imagens, fluxogramas e códigos de programação em Python produzidos e utilizados durante a pesquisa.

2 REVISÃO DA LITERATURA

Neste capítulo será feita uma revisão da literatura dos principais tópicos de importância para essa dissertação, de forma a propiciar uma visão completa e detalhada das tecnologias e métodos que embasam esta pesquisa. Os assuntos abordados serão os seguintes:

- Controle e monitoramento de obras: serão expostas definições sobre acompanhamento de construções, sua importância, e técnicas relevantes para essa etapa da obra.
- Building Information Modelling (BIM): serão apresentados os principais conceitos de BIM, suas dimensões, níveis de desenvolvimento e aplicações para o controle de construções.
- Inteligência artificial (IA): serão apresentados os principais conceitos relacionados à IA, como *machine learning*, *deep learning*, redes neurais artificiais e redes neurais convolucionais. Além disso, serão apresentadas pesquisas que utilizam essas técnicas na construção civil, especialmente para o controle de construções.
- BIM e IA: será apresentada uma breve contextualização da utilização do BIM em conjunto com a IA, bem como as principais aplicações dessas tecnologias na construção civil em geral e no monitoramento de construções.

2.1 CONTROLE E MONITORAMENTO DE CONSTRUÇÕES

O planejamento de uma obra se trata de um processo dinâmico e mutável, além de estar constantemente submetido a imprevistos e a incertezas. Existem diversas razões para que a programação inicial de uma construção seja alterada, sendo que entre elas pode-se citar alterações no projeto, alterações dos métodos construtivos utilizados, mudanças na sequência de execução de algumas atividades, atrasos em fornecimento de material, alterações na produtividade dos trabalhadores, além de um planejamento inadequado ou incompleto (MATTOS, 2019).

Nesse contexto, tem-se que o responsável pela obra necessita constantemente confrontar o que foi previsto com o que foi efetivamente construído, com o objetivo de

avaliar se a obra será finalizada dentro de um prazo adequado ou se serão necessárias providências para que essa meta inicial seja atingida.

A relação entre o planejamento de uma obra e o seu acompanhamento, bem como a importância de cada uma dessas etapas pode ser exemplificada por meio do ciclo PDCA (do inglês, Plan – Planejar, Do – Fazer, Check – Checar e Act – Agir) também conhecido como ciclo de Deming. O ciclo PDCA consiste em uma ferramenta de gestão de processos, buscando sua melhoria contínua através de um processo cíclico, em que é realizado o planejamento de um projeto ou produto, se executa esse planejamento, verifica se o que foi executado está compatível com o que foi concebido inicialmente e, então, se tomam as medidas necessárias para melhorar o processo, voltando-se para a etapa inicial de planejamento (CALÔBA; KLAES, 2016).

Nesse contexto, tem-se que a elaboração do cronograma de uma obra está naturalmente relacionada com a etapa de “planejar”, enquanto a execução da obra se refere à etapa de “fazer”. O monitoramento ou acompanhamento da construção, por sua vez, representa a terceira fase do ciclo PDCA, ou seja, “checar” ou controlar. Por meio dessa checagem ou controle, verifica-se o desempenho do planejamento inicial e, com base nele, podem ser tomadas ações corretivas caso existam grandes alterações ou, no caso de não haver distorções consideráveis, procedimentos para manter o ritmo e a produtividade no canteiro. Esse conjunto de tomada de decisões baseadas no acompanhamento de obra consiste na última etapa do ciclo PDCA - “agir”.

Segundo Mattos (2019), existem três etapas na fase de monitoramento de obras, sendo elas:

- Aferição do progresso das atividades: nesta etapa, a equipe de construção realiza medições com o objetivo de comparar o progresso de cada uma das tarefas previstas no cronograma com o que foi efetivamente executado. Essa medição pode ser tanto por meio de percentual quanto por unidade de medida específica (metros quadrados, metros cúbicos, entre outros).
- Atualização do planejamento: se realiza a efetiva comparação entre o que foi planejado e o que foi realizado. A partir desse confronto, atualiza-se o cronograma da obra.
- Interpretação do desempenho: após a atualização do cronograma da construção, faz-se uma análise se a obra poderá terminar no prazo previsto ou se possui uma tendência a atrasar. Além disso, faz-se uma análise crítica das razões das distorções entre o planejamento e a execução, de forma que se possa

identificar as suas respectivas causas e tomar as medidas corretivas adequadas e mais efetivas.

Mattos (2019) define a coleta dos dados de campo de uma obra para aferição de seu progresso como apropriação. Segundo o autor, existem algumas formas de apropriação, sendo elas: unidades físicas, rateio (percentual), marcos ponderados e por data. A apropriação por unidades físicas, considerado o método mais adequado, ocorre quando a tarefa pode ser medida por meio de quantidades exatas no canteiro, como por exemplo volume de concreto (metros cúbicos).

Nesse sentido, um monitoramento do progresso de construções feito de forma adequada permite que as medidas corretivas necessárias sejam aplicadas durante a execução do projeto, enquanto um monitoramento feito de forma inadequada pode causar diversos atrasos. Dessa forma, identificar uma maneira eficiente de se acompanhar a execução da obra é de fundamental importância, sendo que, atualmente, diversas tecnologias auxiliam na execução dessa tarefa (QURESHI *et al.*, 2022).

Maalek *et al.* (2014) destacaram que, tradicionalmente, os dados sobre a construção no canteiro de obras são coletados de forma manual, sendo uma tarefa trabalhosa, demorada e custosa, especialmente em grandes projetos. Dessa forma, os autores destacaram o uso de diversas tecnologias que podem ser utilizadas para automatizar o monitoramento do progresso de construções, que permitem que o usuário calcule de forma direta o progresso dos trabalhos com os dados capturados, sendo elas a utilização de câmeras, LiDAR (*Light detection and ranging* – tecnologia óptica de detecção remota, como laser) e geradores de imagens tridimensionais. Essas tecnologias possuem como vantagem a melhoria de qualidade das inspeções e gestão da obra, redução de retrabalhos, identificação de situações perigosas e geração de modelos *as built* 3D e 4D da construção.

Rahimian *et al.* (2020) também realizaram um estudo sobre as principais tecnologias utilizadas no monitoramento de obras, mais especificamente as utilizadas para aquisição de dados do canteiro e sua visualização. Os autores destacam o uso dos seguintes tipos de técnicas:

- Geoespaciais: uso de GPSs e do GIS (Geographic Information System – Sistema de Informação Geográfica), banda ultralarga, identificação por radiofrequência e códigos de barras.
- Imagens: escaneamento e laser, fotogrametria e videogrametria.
- Realidade aumentada e realidade virtual.

Entretanto, os pesquisadores afirmam que, apesar do uso das tecnologias geoespaciais e de imagens aprimorarem o acompanhamento das construções, ainda há dificuldades para que elas identifiquem as diferenças entre o que foi construído e o que foi planejado, bem como auxiliar na tomada de decisões com base nos resultados obtidos em campo. Além disso, também é destacado na pesquisa o uso do BIM e de técnicas de aprendizado de máquina nessa tarefa.

2.1.1 Visão computacional aplicada ao monitoramento de construções

Como destacado anteriormente, diversas tecnologias estão sendo empregadas para um monitoramento de construções mais automatizado, menos laborioso e de maior qualidade. Nesse sentido, deve-se ressaltar o uso da visão computacional na execução dessa tarefa.

Segundo Paneru e Jeelani (2021), as áreas de maior aplicação da visão computacional na indústria da construção são a segurança do trabalho e o monitoramento do progresso de obras, seguidos pelo acompanhamento da produtividade e controle de qualidade das construções. O BIM também é bastante utilizado para o acompanhamento de obras, porém ainda é necessário que sejam inseridas informações sobre a construção no modelo tridimensional para que esse acompanhamento seja feito, de forma que o uso de câmeras permite que as informações sejam capturadas com maior facilidade e automatiza a integração desses dados ao modelo tridimensional.

Nesse contexto, Reja *et al.* (2022) identificaram os principais estágios do processo de acompanhamento de obras utilizando a visão computacional, sendo eles:

- 1- Aquisição de dados do que foi construído em campo (*as built*) e conversão dessas informações em um modelo tridimensional por meio de nuvem de pontos.
- 2- Geração do modelo computacional *as built* a partir da nuvem de pontos, utilizando como referência preferencialmente o modelo planejado do projeto.
- 3- Comparação do modelo *as built* com o planejado, de forma a avaliar o progresso da construção tanto de forma visual como por meio de quantificação do trabalho realizado.

Ekanayake *et al.* (2021), por sua vez, conduziram uma revisão da literatura sobre o uso da visão computacional para o controle de construções em ambientes internos. A tarefa apresenta diversos desafios, pois, segundo Kropp *et al.* (2016), as técnicas de visão

computacional utilizadas para monitorar construções em um ambiente externo não podem ser utilizadas em ambientes internos sem uma adaptação, devido a fatores como condições de iluminação insuficientes.

Nesse sentido, os maiores desafios para o acompanhamento de obras em trabalhos internos é a detecção de objetos, as condições de iluminação e o movimento das câmeras. Todavia, é destacado o uso das redes neurais convolucionais para detecção e rastreamento de objetos nesse tipo de tarefa. Também foram encontradas pesquisas que aplicam a visão computacional no controle de construções em ambientes internos na visualização do progresso dos trabalhos por meio de “passeios virtuais” dos modelos, reconhecimento de atividades e captura de informações para criação de modelos BIM *as built* (Ekanayake *et al.*, 2021).

2.2 BUILDING INFORMATION MODELLING

O Building Information Modelling (BIM) pode ser entendido como uma simulação computacional da engenharia e arquitetura, sendo um modelo tridimensional de uma construção com informações e que pode ser utilizado durante todo o seu ciclo de vida, ou seja, desde a fase de projetos até a sua eventual demolição (EASTMAN *et al.*, 2011). A ISO 19650-1 (2018), por sua vez, define o BIM como o uso de uma representação digital e compartilhada de uma construção, de forma a facilitar o design, a construção e a operação da mesma, de forma que se tenha uma base confiável para a tomada de decisões.

O BIM também foi positivado na legislação brasileira em alguns decretos do Governo Federal, como por exemplo o Decreto nº 10.306/2020. Esse dispositivo legal determina o uso do BIM em obras e serviços de engenharia relacionados com a administração pública no âmbito federal. O Decreto define o BIM como sendo um “conjunto de tecnologias e processos integrados que permite a criação, a utilização e a atualização de modelos digitais de uma construção, de modo colaborativo, que sirva a todos os participantes do empreendimento, em qualquer etapa do ciclo de vida da construção” (BRASIL, 2020).

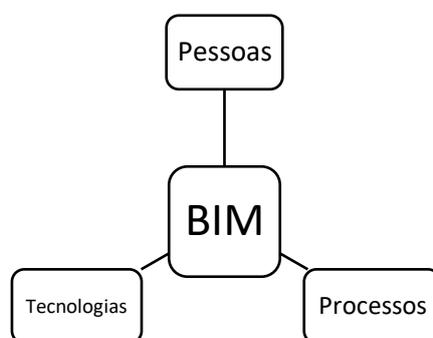
A importância do BIM em obras públicas também foi ratificada pela Nova Lei de Licitações e Contratos (Lei nº 14.133/2021), que estabelece em seu Art. 19, §3º que “Nas licitações de obras e serviços de engenharia e arquitetura, sempre que adequada ao objeto

da licitação, será preferencialmente adotada a Modelagem da Informação da Construção (Building Information Modelling – BIM) ou tecnologias e processos integrados similares ou mais avançados que venham a substituí-la” (BRASIL, 2021).

Nesse sentido, vale destacar que não é somente no Brasil que o governo tem estabelecido incentivos e obrigações à adoção do BIM, mas esse apoio estatal se trata de um fenômeno global (Smith, 2014).

Do exposto, pode-se concluir que o BIM não se trata somente de uma tecnologia ou um conjunto de softwares, mas se trata de todo um processo de se projetar, construir e gerenciar um empreendimento, que envolve tecnologias, pessoas e processos, conforme destacado na Figura 1.

Figura 1: Elementos do processo BIM



Fonte: do autor

O BIM, portanto, se trata de uma nova forma de se projetar, trazendo novos paradigmas para o campo da Arquitetura, Engenharia e Construção (AEC). Por envolver uma equipe multidisciplinar que atua em conjunto desde o começo do projeto, tem-se que o BIM permite que menos erros e retrabalhos ocorram durante a construção, trazendo economia de custos e maior qualidade no desenvolvimento do ativo final.

Outro importante conceito dentro do processo BIM é o de interoperabilidade, que pode ser entendida como a capacidade de troca de dados e informações entre *softwares* distintos sem perdas ou transformações significativas dos dados quando ocorrer essa comunicação entre as plataformas (UTKUCU; SÖZER, 2020). A interoperabilidade é de fundamental importância no processo BIM pois permite que profissionais de diversas áreas trabalhem de forma colaborativa, mas cada um utilizando o melhor *software* ou plataforma de acordo com suas necessidades.

A importância do BIM na indústria da construção civil pode ser destacada quando se analisa o fato de que, a indústria da construção, de forma geral, apresenta inovações em uma velocidade muito menor quando comparada com outros tipos de indústria (Hautala et al., 2017). Um dos principais motivos para essa mudança de paradigmas mais lenta é a falta de dados e de investimentos em tecnologia (FULFORD; STANDING, 2014). Dessa maneira, tem-se que o BIM surge como uma das formas de transformar a construção civil em uma atividade mais digital e tecnológica.

2.2.1 Dimensões do BIM

O BIM, por definição, se trata de um modelo tridimensional digital de uma construção (BIM 3D). Vale destacar, entretanto, que uma simples representação tridimensional da edificação não constitui, por si só, um modelo BIM. Essa representação deve possuir elementos paramétricos, com informações em cada um desses elementos e uma relação entre as representações 2D e o modelo 3D. Informação, segundo o conceito apresentado na ISO 19650-1, são dados representados de uma maneira formalizada, de forma que seja possível a sua interpretação, comunicação e processamento, seja por meios digitais ou por um humano.

Em que pese a base de todo o processo BIM seja o modelo tridimensional anteriormente citado, tem-se que as mais diversas aplicações desse modelo foram relacionadas com “dimensões”, ou seja, o uso desse modelo digital está relacionado a uma “dimensão do BIM” ou “BIM nD” (KOUTAMANIS, 2020).

O BIM 4D ocorre quando se adiciona o parâmetro de tempo ao modelo digital tridimensional, relacionando o modelo 3D ao cronograma de execução da obra. Dessa maneira, quando todos os elementos possuem informações sobre quando serão construídos ou demolidos, pode-se realizar uma simulação passo a passo da construção. Essa dimensão, portanto, reflete o planejamento da obra, permitindo aos participantes do empreendimento criarem cronogramas mais realistas, melhorar a organização do canteiro de obras, aumentar a produtividade dos funcionários com alocações mais inteligentes de recursos e controlar de forma mais assertiva o progresso da construção (PETERS in UNDERWOOD e ISIKDAG, 2009).

No BIM 5D são adicionadas informações de custo aos modelos digitais, o que amplia as possibilidades do modelo tridimensional que já possui recursos para a extração detalhada de quantitativos (CARVALHO e MARCHIORI, 2019). Dessa forma, é

possível que sejam realizados orçamentos mais precisos da construção, com a verificação do custo de cada alternativa de projeto de maneira automatizada. Além disso, ao se juntar as dimensões 4D e 5D do BIM, é possível a criação de cronogramas físico-financeiros mais detalhados e fidedignos, inclusive com simulações em vídeo ou imagens da obra com o respectivo gasto com os elementos já executados (PETERS in UNDERWOOD e ISIKDAG, 2009).

O BIM 6D consiste na associação dos modelos tridimensionais com dados sobre eficiência energética e sustentabilidade. Dessa forma, o BIM pode ser utilizado, por exemplo, para se estimar a quantidade carbono que será emitida pela construção durante seu ciclo de vida (PENG, 2016). Além disso, as informações contidas nos elementos do modelo podem ser utilizadas para análises térmicas da construção, contribuindo para o conforto no interior do edifício (FAN; WONG, 2013).

O uso do BIM para a gestão de instalações (*Facility Management*) se trata do BIM 7D, em que o modelo digital é utilizado para controlar a edificação durante a sua utilização, ou seja, a maior parte do seu ciclo de vida. A sétima dimensão do BIM pode ser aplicada, por exemplo, no controle de manutenções de equipamentos ou partes da edificação (BECERIK-GERBER; CALIS; JAZIZADEH; LI, 2011).

Além das supracitadas dimensões do BIM, Kamardeen (2010) propôs uma oitava dimensão do BIM (BIM 8D), em que se utilizaria a Modelagem de Informação da Construção para se prevenir os acidentes de trabalho que ocorrem na construção civil. O pesquisador propôs um modelo de ferramenta que analisaria o modelo digital e faria uma análise de risco de cada elemento do projeto, classificando o risco em crítico, moderado e baixo. Além disso, a ferramenta proposta faria sugestões de como cada risco poderia ser mitigado ou eliminado, de forma que através de melhorias da execução da construção e do design do projeto final, se prevenisse acidentes de trabalho na construção civil.

O BIM 9D seria uma integração entre o *Lean Construction* e o BIM (Lean-BIM) (ARNAL, 2018), ou seja, conceitos sobre a construção enxuta, redução de desperdícios em canteiros de obras e otimização do fluxo de trabalho aplicados com ferramentas BIM (JAISWAL *et al.*, 2022).

O BIM 10D, por sua vez, trata da construção industrializada utilizada juntamente com o BIM (Ershadi *et al.*, 2021). A ideia é utilizar o BIM para a fabricação de elementos pré-fabricados, que chegam ao canteiro de obras preparados para a instalação. Nesse contexto, haveria um ganho de qualidade na execução e maior controle sobre prazos e custos (ARNAL, 2018).

2.2.2 BIM no controle de construções

O BIM vem sendo constantemente utilizado no controle de construções, seja de forma isolada, seja integrado com Veículos Aéreos Não-tripulados, escaneamento a laser, Internet das Coisas, realidade aumentada ou inteligência artificial.

Álvares (2019) implementou uma metodologia para o monitoramento visual do progresso de construções unindo o BIM, Veículos Aéreos não-tripulados e mapeamento 3D. O método permitiu que o monitoramento fosse feito de forma mais clara e rápida por meio de um código de cores nas imagens computacionais, que permitia facilmente identificar os elementos já construídos, em atraso, adiantados e em execução de acordo com o cronograma. Entretanto, uma das limitações encontradas no trabalho foi a falta de análises visuais para atividades internas.

Cao *et al.* (2018) discutiram e propuseram uma metodologia com a aplicação do BIM com a Internet das Coisas (Internet of Things - IoT) para monitoramento de construções de pontes metálicas, uma vez que esse tipo de projeto é extremamente complexo, dado que envolve a indústria de transformação e a indústria da construção civil, causando problemas de eficiência, custos e monitoramento.

Lin *et al.* (2019) propuseram uma integração entre o BIM 4D e a realidade aumentada para o monitoramento do progresso de construções modulares. A união entre as duas tecnologias permite que possa ser verificado diretamente no canteiro o posicionamento de todos os elementos antes da execução, etapa por etapa e de acordo com o cronograma pré-estabelecido, facilitando a visualização das próximas tarefas a serem executadas no canteiro de obras e evitando assim retrabalhos e consequentes atrasos.

Arif e Khan (2020) assimilaram o modelo BIM com a computação em nuvem para automatizar o controle da evolução de obras e para o cálculo da produtividade no canteiro, sendo uma alternativa de baixo custo para companhias de pequeno e médio porte no Paquistão.

As pesquisas mencionadas permitem concluir que o BIM vem sendo amplamente utilizado na construção civil para o monitoramento de construções de todos os tipos. O uso do BIM nessa área geralmente envolve a comparação do modelo 3D ou da simulação 4D com imagens obtidas por drones ou geradas por escaneamento a laser para se saber quais elementos já foram construídos e se foram construídos ou não com atraso. Nesse contexto, nota-se que usualmente o modelo tridimensional é utilizado como base para

uma comparação com a obra real necessitando que as informações reais da obra sejam inseridas no software BIM para que seja realizado o controle.

2.3 INTELIGÊNCIA ARTIFICIAL

A inteligência artificial pode ser definida como um campo da ciência da computação em que a capacidade humana de raciocinar, resolver problemas, tomar decisões, entre outras tarefas, é simulada por dispositivos computacionais (RUSSEL; NORVIG, 2020). Atualmente a IA está sendo utilizada em diversos campos do conhecimento e sendo aplicada na prática na medicina, engenharia, finanças e vendas. A aplicabilidade da IA é ampla e diversa, sendo utilizada para problemas de regressão, classificação, criação de regras, agrupamentos de elementos semelhantes, visão computacional, processamento da linguagem natural e realização de previsões.

A inteligência artificial está sendo cada vez mais empregada na construção civil, com grande destaque para as diversas técnicas de *machine learning*, *deep learning* e algoritmos genéticos, com aplicação nas áreas de reconhecimento de imagens, otimização de projetos, regressões e classificações (Da Silva; Carvalho, 2022).

O aprendizado de máquina (*machine learning*) pode ser entendido, por sua vez, como um campo da inteligência artificial em que os computadores simulam a capacidade humana de aprender sem serem efetivamente programados (SIMON, 2015), encontrando soluções para problemas a partir da extração de regras e identificação de padrões de um grande conjunto de dados (KOHAVI; PROVOST, 1998). Existem diversas técnicas de aprendizado de máquina, sendo que algumas das principais são: Naive-Bayes, Redes Bayesianas, Árvores de Decisão, Florestas Aleatórias, Máquinas de Vetores de Suporte (SVM) e Redes Neurais.

Existem basicamente três tipos de aprendizado de máquina, conforme os dados disponíveis e a abordagem para a solução dos problemas a serem resolvidos pela máquina: aprendizado supervisionado, não-supervisionado e por reforço.

A aprendizagem de máquina supervisionada é aquela em que existe a figura de um supervisor, que tem o papel de atribuir para cada elemento do conjunto de dados uma classificação ou um valor específico antes da aplicação do algoritmo. Dessa forma, o algoritmo de inteligência artificial irá buscar padrões nesses dados rotulados e aprender com base nas informações fornecidas por esse supervisor. A aprendizagem de máquina

supervisionada, portanto, é aplicada em problemas de classificação e regressão (RUSSEL; NORVIG, 2020).

A aprendizagem não supervisionada, por sua vez, é uma situação em que não há um supervisor, sendo que cada elemento do conjunto de dados é avaliado pelo algoritmo de IA isoladamente, sem ter uma classificação prévia ou um valor correspondente previamente atribuído. O algoritmo por si só irá identificar um padrão. Esse tipo de aprendizagem, portanto, é utilizado para, por exemplo, se agrupar elementos semelhantes ou criar regras de relações entre elementos (Hinton; Sejnowski, 1999).

A aprendizagem por reforço, por fim, trata-se de uma situação em que o algoritmo de aprendizado de máquina adquire inteligência através de iterações com o ambiente, por meio de um sistema de punições e recompensas, aprendendo a executar determinada tarefa (Kaelbling; Littman; Moore, 1996).

Dentro do campo da aprendizagem de máquina, existe a área mais específica de deep learning (aprendizagem profunda), que utiliza várias camadas de processamento para a extração de informações cada vez mais detalhadas dos dados de entrada (DENG; YU, 2014). Um dos principais métodos de aprendizagem profunda usados atualmente são as redes neurais artificiais.

A Figura 2 mostra de forma esquemática a relação existente entre os campos da inteligência artificial, *machine learning* e *deep learning*.

Figura 2: Áreas e subdivisões da inteligência artificial



Fonte: do autor

2.3.1 Redes Neurais Artificiais

As Redes Neurais Artificiais (RNAs) são sistemas computacionais inspirados no funcionamento do cérebro humano, mais especificamente nos neurônios biológicos, sendo utilizadas para a solução de diversos tipos de problemas computacionais. Apesar de as RNAs estarem sendo mais utilizadas nas últimas décadas devido ao aumento da

capacidade computacional, a teoria por trás do neurônio artificial foi desenvolvida em 1943 pelos professores McCulloch e Pitts (McCulloch; Pitts, 1943).

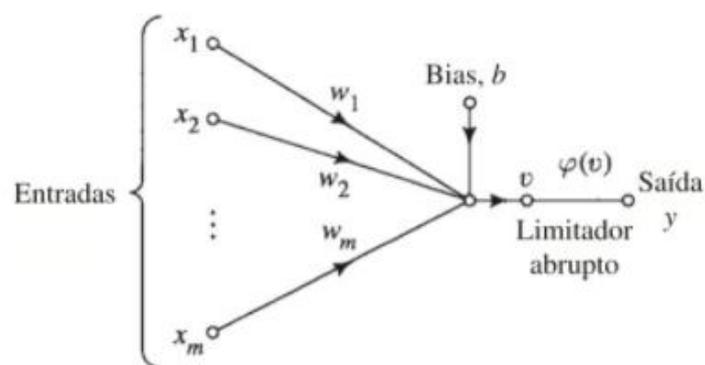
As RNAs são compostas por neurônios artificiais, de forma que, após o sistema passar por um treinamento, ou seja, identificar padrões em dados fornecidos, esses neurônios artificiais possuirão as informações necessárias para a solução de outros problemas semelhantes (HAYKIN, 2001).

Um neurônio artificial está representado na Figura 3, sendo composto por diversos dados de entrada (X_1, X_2, \dots, X_m), sendo que cada entrada possui seu respectivo peso (W_1, W_2, \dots, W_m). O valor de cada entrada é então multiplicado pelo seu respectivo peso e, posteriormente, é realizado o somatório desses produtos. A depender da estrutura da rede neural, pode ser adicionado um valor bias a esse somatório.

Esse somatório então será submetido a uma função de ativação (*activation function*), que se trata de uma transformação nos dados por meio de uma função matemática. O resultado dessa operação, por sua vez, será a saída do neurônio artificial.

Uma vez que se trata de uma rede neural, a saída do resultado da operação da função de ativação no neurônio artificial pode ser transportada para outro neurônio. Dessa forma, a depender do resultado fornecido por esse conjunto de operações, após a *activation function*, o neurônio será ou não ativado, ou seja, a informação de saída será ou não transmitida ao neurônio seguinte conectado.

Figura 3: Exemplo de neurônio artificial

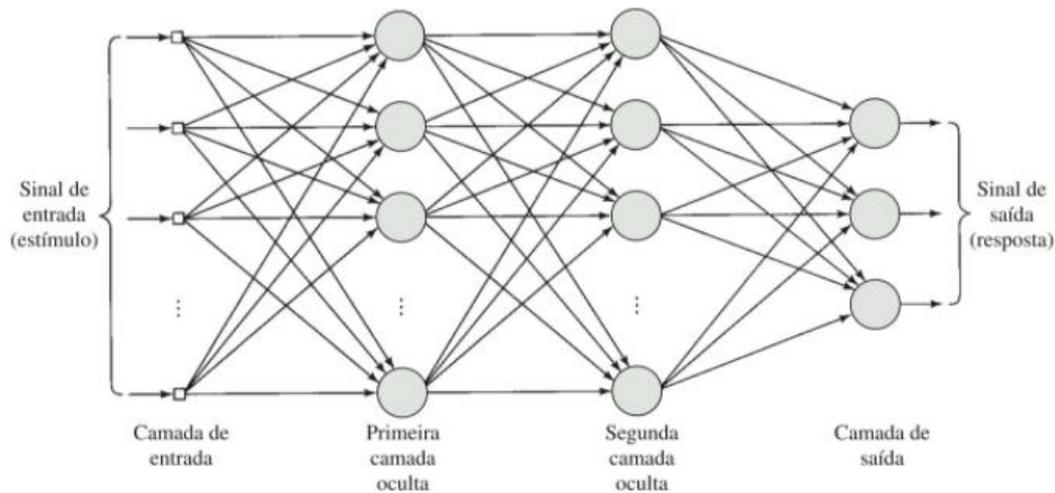


Fonte: HAYKIN (2001)

A redes neurais artificiais são normalmente organizadas em camadas, de forma que os neurônios de uma camada se conectam aos neurônios da camada seguinte, transportando dados. A camada inicial da rede é denominada de camada de entrada,

enquanto a última camada é denominada de camada de saída. Pode haver diversas camadas intermediárias na rede neural. A combinação de vários neurônios artificiais em diversas camadas dá origem a uma rede multicamada, que são utilizadas para a solução de problemas mais complexos, conforme mostrado na Figura 4.

Figura 4: Exemplo de rede neural



Fonte: HAYKIN (2001)

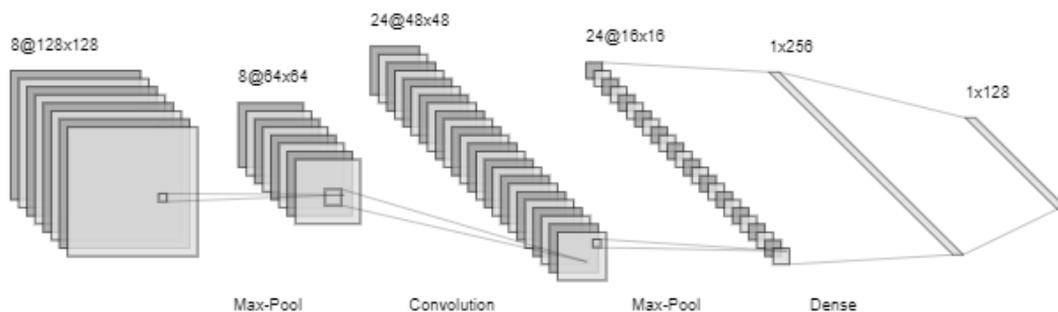
O aprendizado de uma rede neural é iniciado quando se atribui valores aleatórios para os pesos de cada neurônio artificial. Uma base de dados pré-estabelecida de treinamento será utilizada para o aprendizado da RNA, fornecendo dados de entrada para a mesma. Com os dados de entrada da rede neural e os pesos aleatórios previamente estabelecidos, o resultado da rede e seu respectivo erro são calculados. Novos pesos são então calculados conforme o erro previamente obtido e a rede neural é atualizada com os novos pesos, em um processo conhecido como *backpropagation*. Esse procedimento é repetido por um número de vezes determinado pelo programador ou até que um valor pequeno de erro pré-estabelecido seja obtido. Ao fim desse processo, a RNA estará treinada, ou seja, os pesos de cada neurônio artificial obtidos nesse processo iterativo poderão ser utilizados para a classificação ou regressão de dados até então desconhecidos.

Uma das maneiras de validar os resultados da rede neural e verificar se a mesma está apta para ser utilizada é submeter uma base de dados de teste a essa rede, verificando se os resultados reais são compatíveis com os obtidos pela rede neural.

2.3.2 Redes Neurais Convolucionais

Uma Rede Neural Convolutiva (RNC) é um tipo de Rede Neural Artificial utilizada principalmente para a análise de imagens (VALUEVA; NAGORNOV; LYAKHOV; VALUEV; CHERVYAKOV, 2020). Uma RNC é capaz de captar as principais características de uma imagem e diferenciá-la de outra, com um pré-processamento menor quando comparado com outros algoritmos de classificação de imagens (XU; ZHANG, 2022) (AHILA PRIYADHARSHINI *et al.*, 2019). Em que pese seja majoritariamente utilizada para problemas de classificações, também é possível fazer regressões com redes neurais convolucionais, ou seja, calcular o valor de uma variável dependente com base nos valores de variáveis independentes. Um esquema de uma rede neural convolutiva pode ser verificado na Figura 5.

Figura 5: Exemplo de rede neural convolutiva



Fonte: <<http://alexlenail.me/NN-SVG/LeNet.html>>

As redes neurais convolucionais apresentam neurônios artificiais assim como as redes tradicionais, a diferença, entretanto, são as camadas de que realizam operações de convolução e de *pooling*.

A operação de convolução ocorre através de uma série de filtros que são aplicados sobre a imagem original ao longo de sua altura e comprimento, gerando uma série de transformações nessa imagem para que seja submetida à camada seguinte. No início da etapa de treinamento de uma rede neural convolutiva, esses filtros, também chamados de *kernels*, iniciam com valores aleatórios, entretanto, durante o treinamento, os valores dos filtros vão sendo atualizados, tendo como resultado *kernels* capazes de extrair as características mais importantes de uma imagem, bem como sua posição espacial na

imagem. O resultado de uma imagem depois de ser submetida a esses filtros, que será então a entrada para a próxima camada da rede neural convolucional, é comumente denominado de mapas de características ou *feature maps*. A quantidade de *feature maps* gerados dependerá da quantidade de filtros que serão treinados.

A operação de *pooling*, por sua vez, reduz o tamanho dos dados de entrada na camada de *pooling*, tendo como saída dados com uma menor dimensão. Essa redução das dimensões dos dados de entrada ocorre por meio de uma função *pooling*, que substitui todos os elementos de uma região por um único elemento obtido por meio de operações estatísticas a partir dos dados originais. Dessa forma, uma camada de *pooling* pode receber como entrada uma imagem de grandes dimensões, mas, após uma série de operações nessa imagem, um agrupamento de pixels dessa imagem é substituído por um único pixel, cujo valor será resultado de uma operação matemática dos pixels que ele está substituindo. A saída da camada será, portanto, uma imagem com menores dimensões.

Existem duas principais operações de *pooling*: *Max pooling*, que diminui os dados de uma região pelo dado de maior valor (Zhou *et al.*, 1988), e o *average pooling*, que substitui os dados originais por sua média aritmética.

Dessa forma, as camadas de convolução e de *pooling* são capazes de realizar modificações nas imagens originais submetidas a uma RNC, de forma a captar suas principais características e reduzir suas dimensões para um menor processamento de informações pelo algoritmo. Esses resultados são então transferidos a uma rede neural densa, que então prossegue para solucionar problemas de classificação ou de regressão nas imagens.

2.3.3 Redes Neurais Artificiais na construção civil

Akinosho *et al.* (2020) conduziram uma revisão da literatura sobre a aplicação do aprendizado profundo (redes neurais com várias camadas) na construção civil. A pesquisa explorou o uso de diferentes arquiteturas de redes neurais, como as redes neurais profundas, redes neurais convolucionais e redes neurais recorrentes, chegando ao entendimento que as principais aplicações desse tipo de algoritmo na construção civil são o monitoramento da estrutura das obras, segurança do trabalho, reconhecimento de atividades no canteiro, simulações de performance, modelagem de ocupação de edifícios, previsão de demandas energéticas de edificações, previsão do custo de construções e classificação de modelos BIM.

Khallaf e Khallaf (2021), por sua vez, também conduziram uma revisão sistemática da literatura sobre a aplicação do aprendizado na construção civil e identificaram sua utilização no rastreamento de equipamentos, detecção de fissuras em construções, gestão de obras, avaliação de redes de esgoto, aprimoramento de nuvens de pontos tridimensionais, entre outros tópicos. Na pesquisa, também foram abordados os desafios do uso do aprendizado profundo na construção civil, sendo citados o tamanho dos bancos de dados, o fato de um modelo treinado com um objetivo específico não poder ser utilizado em um problema distinto, a qualidade das câmeras para capturar imagens em construções, entre outros aspectos.

Um estudo realizado por Mostafa e Hegazy (2021) identificou que a análise de imagens na construção civil é utilizada principalmente em problemas de segurança no canteiro, monitoramento de obras e identificação de danos. Os pesquisadores também concluíram que a principal técnica utilizada em pesquisas nessa área são as redes neurais convolucionais, que, para obterem resultados satisfatórios, é necessário o uso da aprendizagem supervisionada e uma grande quantidade de imagens para o aprendizado. Além disso, a necessidade de câmeras de alta-resolução foi enfatizada, de forma que se possa obter o máximo dos benefícios da visão computacional.

Nesse sentido, pode-se destacar algumas pesquisas conduzidas no campo da engenharia de construções com a aplicação de redes neurais artificiais, tanto para solução de problemas de classificação como de regressão.

Um exemplo de aplicação de redes neurais artificiais em problemas de regressão é a pesquisa de Göymen *et al.* (2021), em que se compararam diversos algoritmos de *machine learning*, entre eles as redes neurais artificiais, para a previsão da resistência do concreto, a partir de alguns dados de entrada, como por exemplo o teor de umidade, a velocidade do pulso ultrassônico e a umidade relativa do concreto endurecido. O algoritmo que forneceu os melhores resultados nessa pesquisa foram as árvores de decisão.

Outro exemplo de redes neurais aplicadas a problemas de regressão é a pesquisa de Barros (2019), que utilizou o algoritmo para a estimativa de custos de construção de rodovias, utilizando para o treinamento da rede os dados de projetos do Departamento Nacional de Infraestrutura e Transporte (DNIT). Após o treinamento da rede neural, ao se comparar o custo real histórico dos projetos e o custo obtido pela regressão feita pela RNA, foi calculado um erro médio de 9%, o que demonstra a grande aplicabilidade das RNAs para estimativas de custos na construção civil.

Um dos exemplos do uso de RNCs para o reconhecimento de imagens aplicado à gestão do canteiro de obras é exemplificado por He *et al.* (2020), em um estudo em que foram utilizadas redes neurais convolucionais para a detectar o uso de capacetes de segurança em obras, em imagens obtidas por câmeras de monitoramento. A RNCs utilizada obteve uma taxa de acertos superior a 90%.

Outro uso de redes neurais no reconhecimento de imagens é a pesquisa realizada por Fei *et al.* (2021), em que RNCs foram novamente utilizadas para classificar tipos de pavimentos, novamente com taxas de acerto superiores a 90%.

Shanti *et al.* (2022) realizaram uma pesquisa em que foi desenvolvido um sistema com integrou o aprendizado profundo com a utilização de drones para monitorar trabalhadores em tempo real quando estivessem realizando trabalhos em altura. O drone era utilizado para capturar imagens e vídeos da construção, de forma que um modelo pré-treinado identificava eventuais falhas ou violações de segurança, em diferentes condições de iluminação e de tempo. O sistema apresentou uma precisão de 90% e detectou em média uma falha de segurança a cada 12 segundos.

Do exposto, pode-se concluir que as redes neurais vêm sendo amplamente utilizadas na construção civil. Redes neurais comuns são frequentemente utilizadas para pesquisas envolvendo regressões, ou seja, quando se há um grande conjunto de dados e utiliza-se as redes neurais para estabelecer e identificar uma relação entre eles, de forma que se possa avaliar dados novos a partir das relações encontradas. As redes neurais convolucionais, por sua vez, também são amplamente utilizadas para o rastreamento de objetos e classificação de imagens, com diversas aplicações. Entretanto, ainda não há pesquisas de destaque que utilizam as redes neurais convolucionais para se fazer regressões no campo da construção civil.

2.4 BIM E IA

O uso do BIM juntamente com a inteligência artificial foi abordado por Eastman *et al.* (2008), sendo proposto que tecnologias como *deep learning* sejam aplicadas integradas com plataformas BIM para a checagem de normas, revisão da qualidade de projetos, entre outros usos. É destacado que, com o aumento do número de projetos utilizando a metodologia BIM, algoritmos de aprendizagem de máquina (*machine*

learning) serão cada vez mais aplicados, uma vez essas técnicas necessitam de um universo de dados para serem aplicadas.

Da Silva e Carvalho (2022) fizeram um Mapeamento Sistemático da Literatura (MSL) com o objetivo de investigar as produções científicas que integrassem o BIM e a Inteligência Artificial. Nesse contexto, foi descoberto que essa combinação se trata de um campo de pesquisa em expansão, sendo que as técnicas de IA mais combinadas com o BIM são as redes neurais artificiais, os algoritmos genéticos e as Máquinas de Vetores de Suporte (em inglês, Support Vector Machines - SVM). Entre as aplicações dessas pesquisas, estão principalmente as áreas de sustentabilidade das construções, custos e otimizações de projetos.

Zabin *et al.* (2022) também conduziram uma revisão sistemática da literatura sobre as aplicações do aprendizado de máquina com o BIM. Na pesquisa, identificou-se um aumento significativo de artigos publicados entre 2010 e 2021 sobre o uso conjunto da integração BIM-IA, abordando todas as fases de projetos. Todavia, foi na fase de concepção dos projetos que houve o maior número de pesquisas publicadas, sendo que as técnicas de aprendizado de máquina mais utilizadas foram as redes neurais e mineração de dados, enquanto o esquema IFC foi o padrão de dados mais mencionado. Na fase de desenvolvimento de projetos, as aplicações mais comuns foram a automação do planejamento e desenvolvimento do projeto, detecção de interferências, análise de produtividade do projeto, checagem automática de regras, predição de performance energética e análise de descarte de materiais, enquanto as técnicas mais utilizadas foram as Máquinas de Vetores de Suporte, seguidas pelas redes neurais e mineração de dados. Na fase de construção, por sua vez, as aplicações do BIM-IA mais comuns foram a automação da geração de cronogramas, monitoramento de construções, automação e melhoria de estimativas de custos e criação de sistemas inteligentes para identificação de riscos.

Como exemplos de aplicações da integração BIM-IA, Farinha *et al.* (2018) utilizaram Support Vector Machines (SVM) e redes neurais para classificar imagens de modelos BIM em três categorias: edifício de apartamentos, construções industriais e outros. As redes neurais treinadas neste estudo obtiveram uma taxa de acerto superior a 89%, enquanto as SVM obtiveram uma precisão de 57%. Dessa forma, uma das conclusões deste trabalho é que as redes neurais artificiais podem ser utilizadas para a classificação de imagens artificiais de um modelo geradas por um software BIM.

Jung *et al.* (2020) também utilizaram técnicas de deep learning para a classificação de imagens computacionais provenientes de um modelo BIM. Foram treinadas redes neurais convolucionais com o intuito de classificar subtipos de elementos BIM, particularmente portas e paredes, com uma acurácia de 92% e 95%. A alta taxa de acerto nesse caso pode ser explicada pois as redes neurais treinadas no estudo puderam identificar características geométricas locais de cada um dos elementos e utilizar esses detalhes para distinguir os elementos.

Entretanto, tem-se que associação do BIM com a inteligência artificial também tem sido amplamente utilizada no campo do controle do progresso de construções, como pode ser destacado em algumas pesquisas.

Golparvar-Fard *et al.* (2010), por exemplo, buscaram automatizar o reconhecimento do progresso físico de construções. Primeiramente, buscaram reconstruir o as-built da construção como um modelo BIM 4D. Posteriormente, um modelo BIM-IFC era integrado a esse as-built, de forma que um modelo de aprendizado de máquina baseado em probabilidade Bayesiana foi utilizado para detectar de forma automática o progresso físico da obra.

Deng *et al.* (2020), por sua vez, propuseram uma integração entre BIM e visão computacional para monitorar o progresso de trabalhos internos de construção, nesse caso, a instalação de azulejos. A primeira fase da pesquisa foi treinar um algoritmo SVM para reconhecer azulejos, sendo que, posteriormente, uma câmera foi utilizada para reconhecer a área de azulejos já instalados de forma automatizada. Esse progresso, por sua vez, era transferido a um modelo BIM hospedado na nuvem.

Pan *et al.* (2021) realizaram uma pesquisa integrando BIM, Internet das Coisas e mineração de dados, com o intuito de desenvolver um gêmeo digital para controle e aprimoramento de processos de construção. Nesse sentido, as informações capturadas por meio da Internet das Coisas são mapeadas para arquivos no sistema IFC, de forma que, por meio de mineração de dados, é possível identificar diferenças entre o comportamento real e o planejado. Dessa forma, foi possível gerar um as-built em formato IFC, de forma que fosse possível monitorar o progresso de construções.

Braun *et al.* (2019) propuseram um modelo que utiliza o BIM 4D, redes neurais artificiais e fotogrametria para automatizar a classificação de imagens e elementos do canteiro de obras. Nessa pesquisa, as imagens obtidas por fotografias do canteiro de obras foram sobrepostas às imagens do modelo BIM, de forma que fosse possível identificar quais elementos já foram construídos e quais ainda não foram. Como o modelo

computacional possui as classificações de cada um dos elementos modelados (fôrmas, pilares, lajes, entre outros), foi possível se classificar os elementos do canteiro, com uma assertividade superior a 91%, automatizando um trabalho que anteriormente durava várias horas.

Fang *et al.* (2020) desenvolveram uma metodologia baseada em visão computacional que, a partir de vídeos de uma construção, é capaz de identificar e monitorar automaticamente o progresso da construção de paredes pré-fabricadas. Após a identificação de quais paredes foram efetivamente erguidas, essa informação é enviada ao software BIM (Revit) através de um plug-in com a respectiva informação de data e hora da instalação da parede, de forma que foi possível monitorar em tempo real o progresso da construção. O estudo também identificou que algumas das limitações para a completa implementação dessa metodologia são a posição e qualidade das imagens da câmera de vídeo, a necessidade de um hardware capaz de processar a quantidade de imagens geradas.

Rahimian *et al.* (2020) utilizaram a plataforma de desenvolvimento de jogos Unity com o intuito de integrar informações provenientes de modelos BIM e imagens da construção as-built, as quais foram processadas com técnicas de visão computacional para sobrepor imagens reais da construção com o modelo planejado. Dessa maneira, o modelo tridimensional computacional é automaticamente atualizado com a condição atual da construção.

Nesse contexto, nota-se que o uso conjunto da inteligência artificial e do BIM já é uma realidade, com diversas pesquisas tratando dessa integração. As aplicações são diversas, como o desenvolvimento de projetos, planejamento da obra e orçamentação. Além disso, o BIM se mostra compatível com o uso de diversas técnicas de IA, como os algoritmos genéticos, as máquinas de vetores de suporte (SVM) e as redes neurais artificiais.

Com relação ao campo da visão computacional e das redes neurais convolucionais, nota-se que os algoritmos de IA já conseguem interpretar os elementos do modelo BIM e relacioná-los com os elementos reais de uma obra e vice-versa, ou seja, imagens do software podem ser utilizadas como imagens reais para simulações e análises. Entretanto, a maioria dos estudos envolve apenas a identificação e classificação dos elementos, faltando pesquisas que analisam imagens e, a partir das mesmas, fazem cálculos e regressões para a obtenção de valores numéricos do progresso da construção.

2.5 Síntese da Revisão da Literatura

O BIM consiste em um processo transformador na área da construção civil, com aplicações na área de desenvolvimento de projetos, durante a construção e até mesmo no pós-obra. Particularmente na área de controle e monitoramento de construções, tem-se um uso amplo do BIM, principalmente como uma ferramenta de visualização do status atual da obra, permitindo uma gestão mais eficiente do cronograma da construção, que deixa de ser um conjunto extenso de linhas, tabelas e gráficos para se tornar um conjunto de imagens ou até mesmo um vídeo com detalhes e informações de mais fácil assimilação. Também vale destacar que o BIM é um processo que é frequentemente integrado com outras tecnologias para soluções de problemas, como a inteligência artificial, a visão computacional, a realidade aumentada, a internet das coisas e o escaneamento a laser. Esse potencial de associação do BIM com outras tecnologias se deve a diversos fatores, sendo que um dos principais é a capacidade do modelo tridimensional computacional de armazenar informações, o que permite que dados obtidos por outras plataformas sejam transportados para dentro de softwares BIM.

A inteligência artificial, por sua vez, também é amplamente utilizada na construção civil. Dentre os diversos algoritmos e técnicas de IA que podem ser empregados no campo da engenharia civil, há um grande destaque para as redes neurais, devido a sua grande capacidade de solucionar problemas de classificação e de regressão. Salienta-se o potencial desse algoritmo de encontrar padrões em dados históricos e armazenar essas informações em seus neurônios e camadas em um modelo treinado, de forma a solucionar problemas semelhantes com dados novos. Um tipo específico de redes neurais são as redes neurais convolucionais, que possuem a capacidade de trabalhar com imagens através de operações como a convolução e o *pooling*, sendo um importante algoritmo na área de visão computacional.

As RNCs são utilizadas na área da construção civil em diversos contextos, mas ainda há poucos estudos que trabalhem o uso das RNCs com o BIM. Além disso, vale ressaltar que as RNCs são utilizadas majoritariamente em problemas de classificação de imagens, apesar de, assim como as redes neurais comuns, poderem ser utilizadas em problemas de regressão. Dessa forma, essa revisão da literatura permitiu identificar uma lacuna no conhecimento científico tanto na integração entre o BIM e as RNCs, como no uso das RNCs em problemas de regressão de forma geral.

Por fim, essa revisão da literatura permitiu identificar que, embora diversas tecnologias sejam aplicadas no monitoramento de obras, quando se trabalha com duas ou várias tecnologias distintas, nem sempre há uma troca de dados e informações de forma ideal entre elas. Dessa forma, ratifica-se a importância de estudos aprofundados sobre as diversas associações de técnicas para o monitoramento de construções, de maneira a potencializar o uso umas das outras.

Do exposto, tem-se que essa pesquisa, ao estudar a relação entre o BIM e as redes neurais convolucionais para regressão, busca explorar e preencher uma lacuna do conhecimento científico na área da construção civil.

3 METODOLOGIA

Este capítulo tem como objetivo explicar e detalhar como a pesquisa será conduzida, de forma a atender os objetivos gerais e específicos do trabalho.

De forma resumida, a pesquisa será desenvolvida seguindo as seguintes etapas:

1 – Desenvolvimento de um banco de dados de imagens de uma simulação BIM 4D com o respectivo progresso da construção

Nessa etapa irá se partir do projeto estrutural desenvolvido em plataforma BIM, fornecido em formato IFC, e então serão modelados elementos do canteiro de obras, como fôrmas e bandejas. Realizada essa etapa, o modelo BIM tridimensional será integrado com o cronograma da construção de forma a se gerar uma simulação da construção (BIM 4D), sendo que essa simulação consistirá em um conjunto de imagens sequenciais da obra. Posteriormente, será calculado o progresso da construção dos elementos estruturais para cada imagem do banco de dados. O produto dessa etapa será, portanto, um banco de dados com imagens de uma simulação BIM 4D com diversas etapas da obra em diversos ângulos diferentes, sendo que cada imagem estará vinculada ao respectivo progresso da construção na respectiva imagem, considerando a evolução da implantação dos elementos estruturais.

2 – Desenvolvimento da estrutura e treinamento das redes neurais convolucionais

Nessa etapa da pesquisa serão definidas as estruturas das redes neurais convolucionais e será realizado o treinamento do algoritmo de aprendizado de máquina selecionado. A primeira fase será definir uma configuração para as redes neurais convolucionais trabalharem com problemas de regressão. Para tanto, serão avaliadas estruturas tradicionais desse algoritmo de visão computacional e, a partir delas, realizar as adaptações necessárias para que imagens possam ser submetidas ao algoritmo de inteligência artificial e o mesmo passe a gerar valores contínuos a partir desses dados de entrada, ou seja, realize uma regressão. Nesse contexto, para que também seja possível se avaliar a influência dos hiperparâmetros das redes neurais convolucionais nos resultados finais do algoritmo, serão gerados 3 modelos distintos, com variações no número de neurônios e camadas.

O próximo passo será subdividir o banco de dados gerado na etapa anterior em dados de treinamento e dados de teste, sendo que os dados de treinamento serão então

utilizados para o treinamento da rede neural convolucional. Conforme destacado anteriormente neste trabalho, o treinamento da RNC consiste em um processo de atualização de seus pesos baseado nas imagens fornecidas e o respectivo progresso em cada uma, utilizando o método de *backpropagation*. Espera-se nesse processo, portanto, que as redes neurais convolucionais consigam captar as principais características das imagens fornecidas e fazer uma correlação com o respectivo progresso da obra disponível na imagem.

3 – Avaliação estatística das redes neurais convolucionais geradas

A fase de avaliação estatística das redes neurais convolucionais consiste em submeter o banco de dados gerado na primeira etapa (tanto as imagens de treinamento quanto as imagens de teste) às redes neurais convolucionais treinadas, de forma a obter uma estimativa do progresso da construção dos elementos estruturais na respectiva imagem. Serão utilizadas então duas métricas estatísticas, como por exemplo o Erro Médio Absoluto (em inglês, *Mean Absolute Error* - MAE) e a Raiz Quadrática Média (*Root Mean Squared Error* - RMSE) para avaliar a precisão dos resultados gerados pelo algoritmo de aprendizado de máquina. A principal razão de se avaliar o modelo utilizando tanto os dados de treinamento quanto os dados de teste é para que se possa verificar também a ocorrência de *underfitting* e *overfitting* nos modelos após o treinamento.

4 – Análise dos *feature maps* gerados pela rede neural convolucional

Após o treinamento e avaliação estatística das RNCs, a próxima etapa será uma avaliação dos mapas de características (*feature maps*) gerados pelos modelos quando se submete uma imagem para regressão. O objetivo desta etapa é buscar compreender quais foram as características que o algoritmo de aprendizado de máquina identificou nas imagens de treinamento que possibilitam realizar a regressão para calcular o progresso da construção dos elementos estruturais. A partir dessa análise, pode-se verificar pontos de melhoria no desenvolvimento de modelos futuros.

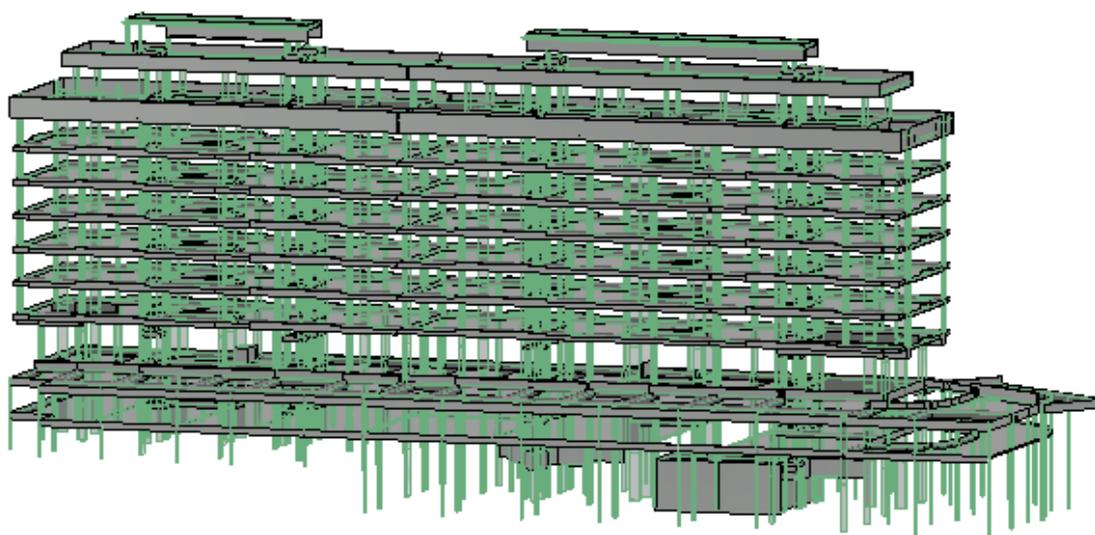
Feita esta breve análise de cada etapa da pesquisa, abaixo seguem os procedimentos mais detalhados, bem como uma descrição de cada ferramenta utilizada.

3.1 Parte 1 – Desenvolvimento de um banco de dados de imagens de uma simulação BIM 4D com o respectivo progresso da construção

Para o início do desenvolvimento dessa pesquisa, foram fornecidos o projeto estrutural de um edifício residencial de 6 pavimentos em um modelo BIM tridimensional, o cronograma dessa construção e seu respectivo orçamento com quantitativos. O projeto estrutural foi fornecido através de um modelo IFC inserido no software Autodesk Revit, enquanto o cronograma foi disponibilizado em um arquivo do MS Project e o orçamento no MS Excel.

Como o objetivo dessa etapa é criar um banco de dados com imagens das etapas da construção, a primeira etapa foi modelar o canteiro de obras em um software BIM, de maneira que esses elementos também possam fazer parte da simulação BIM 4D. Para tanto, foi utilizado um *template* do Autodesk Revit com famílias BIM específicas para desenvolvimento de projetos de canteiro e segurança de trabalho, como por exemplo famílias de fôrmas, escoramentos e bandejas. Dessa forma, iniciou-se um novo projeto no Autodesk Revit utilizando-se como padrão esse *template*. Após o início do projeto no software, o modelo estrutural em IFC do edifício foi inserido no projeto, de forma que pudesse ser utilizado como base para modelagem das fôrmas dos pilares e bandejas, como pode ser verificado na Figura 6. Vale ressaltar, portanto, que a modelagem dos elementos do canteiro de obras foi feita com base no projeto estrutural do edifício já disponibilizado.

Figura 6: Modelo estrutural inserido no software Autodesk Revit



Dessa forma, como destacado anteriormente, para se obter uma simulação da construção mais próxima da real, além do modelo estrutural, serão inseridas famílias de elementos do canteiro de obras. Apesar do *template* utilizado possuir diversas famílias disponíveis de elementos temporários da construção, optou-se por modelar somente as fôrmas dos pilares a partir do andar térreo e as bandejas do primeiro e quarto pavimentos. Entre as razões para a modelagem somente desses elementos do canteiro, está o fato de que, caso fossem inseridos outros objetos BIM, como por exemplo linhas de vida e escoramentos, o modelo final seria demasiado pesado e de difícil manipulação, de forma que se considerou que a modelagem somente das bandejas e das fôrmas seriam suficientes para uma eficaz manipulação dos arquivos, além de fornecerem uma compreensão suficiente do andamento do projeto a partir das imagens geradas pela simulação. Nesse sentido, cabe destacar a definição de *over-modelling* no contexto de modelos BIM, que consiste em um excesso de atenção na modelagem geométrica, com pouca ou nenhuma informação associada a esses elementos, de forma que uma grande quantidade de detalhes desnecessários resultam em um modelo “pesado” de ser manipulado (Kouider *et al.*, 2019). Dessa forma, a opção por inserir no projeto apenas alguns componentes do canteiro é uma forma de evitar o *over-modelling*, buscando-se inserir apenas elementos que irão contribuir com informações relevantes para a simulação da construção.

As bandejas do primeiro e quarto pavimentos foram modeladas no entorno da estrutura de concreto, conforme fotografias da obra real. As fôrmas dos pilares, por sua vez, foram inseridas no projeto em cada pilar do modelo, sendo que as dimensões desses elementos foram obtidas a partir dos parâmetros dos objetos IFC de cada um dos pilares. Inseridos esses elementos no modelo estrutural base, foi obtido um modelo simplificado em BIM do canteiro de obras. As famílias de fôrmas de pilares e de bandejas de proteção podem ser verificadas nas Figuras 7 e 8.

Figura 7: Fôrma de pilar utilizada no projeto

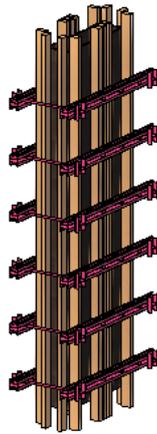
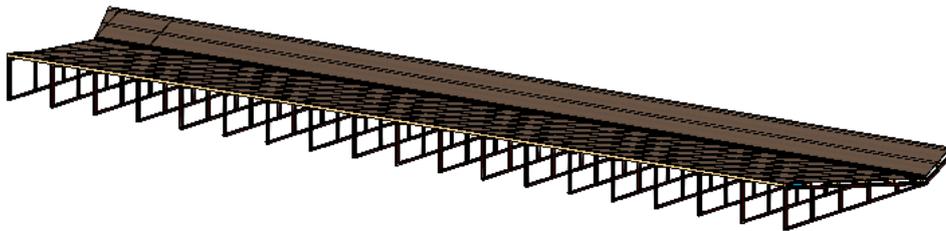


Figura 8: Família de bandeja utilizada no projeto



A próxima etapa da criação do banco de dados é a geração da simulação da construção (BIM 4D). Para tanto, optou-se pelo uso do software Autodesk Navisworks, que se trata de uma plataforma de integração de dados de projeto, permitindo a simulação da obra, controle de custos da construção e identificação de interferências entre projetos (Autodesk, 2023).

Para que seja feita a simulação, o modelo do Autodesk Revit foi inserido no Navisworks a partir do comando “Inserir” (em inglês, *Append*), conforme mostra a Figura 9. A Figura 10 mostra alguns dos tipos de formatos de arquivos suportados pelo Navisworks, mostrando a interoperabilidade que esse software possui com outras plataformas.

Figura 9: Comando *Append* no Autodesk Navisworks

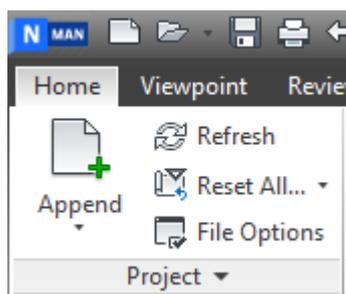
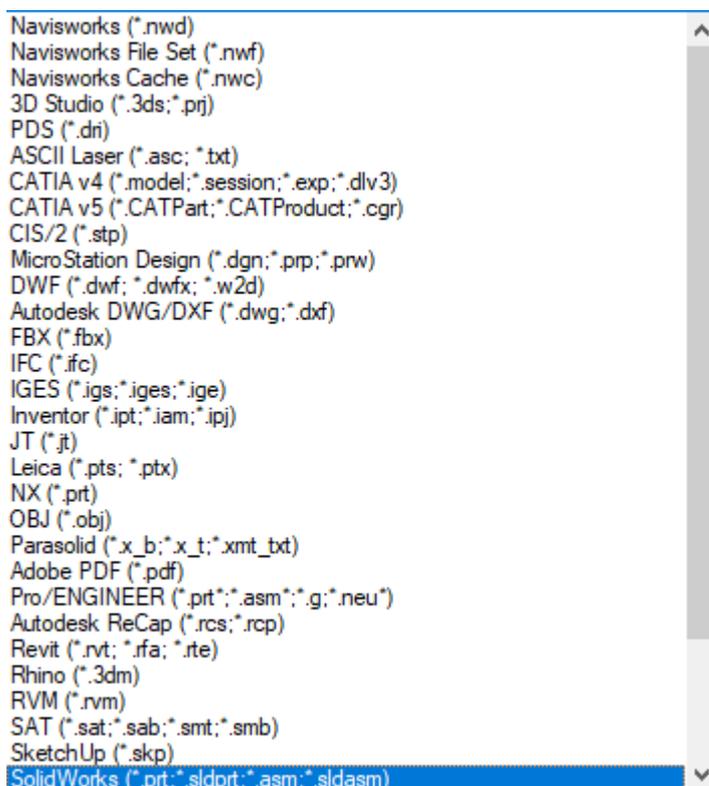


Figura 10: Lista de formatos suportados pelo Autodesk Navisworks

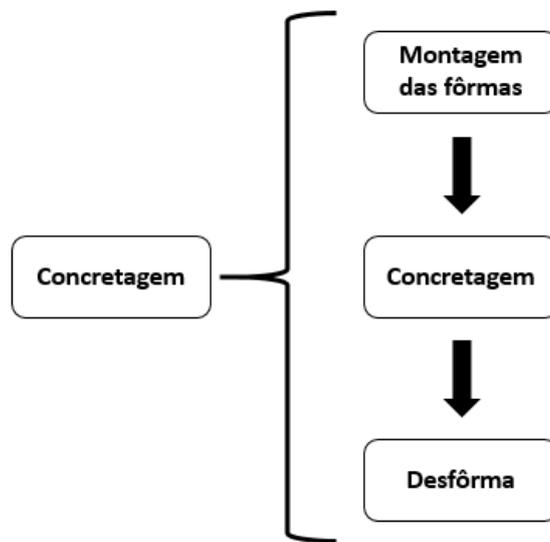


Em seguida, o cronograma da construção será inserido no Autodesk Navisworks para ser integrado ao modelo BIM tridimensional, de forma a gerar a simulação. Como nesse projeto somente se está avaliando o progresso da construção dos elementos estruturais, as partes do cronograma não relacionadas com essa disciplina foram excluídas, de forma que o cronograma original, que tinha 1232 tarefas, foi reduzido para 81, ou seja, somente 6,57% do planejamento original.

Além da exclusão de tarefas não relacionadas com a construção de elementos estruturais, a etapa referente à concretagem dos pilares foi subdividida em 3 subtarefas, que consiste nas etapas de montagem das fôrmas, a efetiva concretagem dos pilares e a

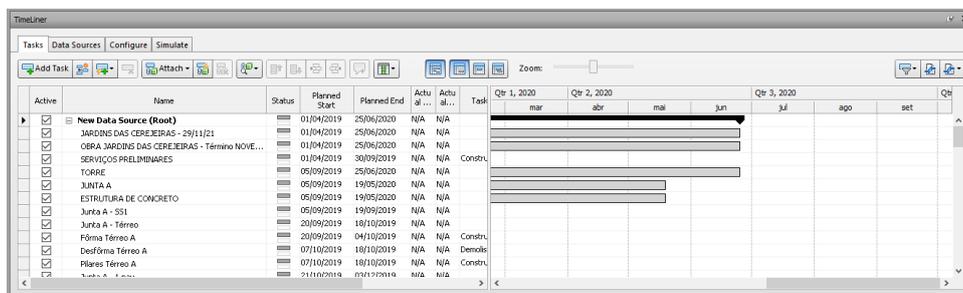
posterior desfôrma. O objetivo dessa subdivisão é criar um maior número de etapas na simulação da construção, de forma que sejam gerados um maior número de imagens e a simulação BIM 4D seja mais detalhada. Para se estimar o prazo de cada uma dessas tarefas, tomou-se o prazo da tarefa original e dividiu-se esse prazo por 3, de forma que cada uma das novas etapas tivesse a mesma duração, e a duração total fosse igual à da tarefa original. Uma esquematização das modificações feitas nessa parte do cronograma pode ser verificadas na Figura 11.

Figura 11: Detalhamento das modificações realizadas no cronograma da construção



Uma vez que o cronograma está pronto e com as modificações necessárias, o mesmo é inserido no Autodesk Navisworks, através da ferramenta *TimeLiner*, conforme mostra a 12.

Figura 12: Ferramenta *TimeLiner* do Autodesk Navisworks



Fonte: do autor

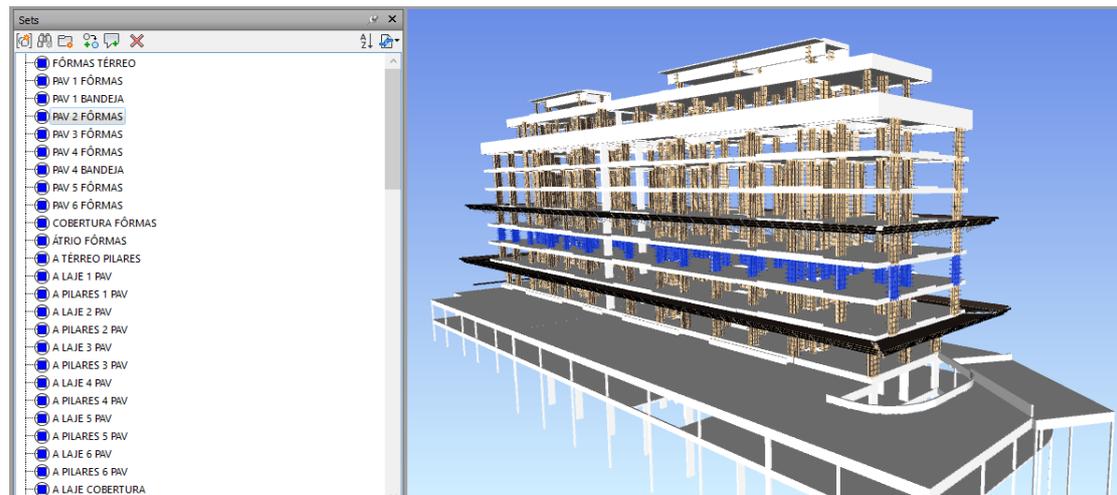
Por fim, dentro da ferramenta *TimeLiner*, define-se se as tarefas serão do tipo “construção” (*construction*), “demolição” (*demolition*) ou “temporário” (*temporary*). As tarefas do tipo “construção” fazem com que os elementos selecionados surjam durante a simulação da construção e não desapareçam posteriormente. As tarefas do tipo “demolição” fazem com que os elementos selecionados, que estão visíveis no começo da simulação, desapareçam quando se iniciar a simulação de certa atividade. Por fim, as tarefas “temporárias” fazem com que os elementos selecionados apareçam na simulação quando se iniciar certa atividade do cronograma e desapareçam quando essa mesma tarefa se encerrar. Todas as tarefas do cronograma inserido no Autodesk Navisworks serão do tipo “construção”, exceto as tarefas de desfôrma, que serão do tipo “demolição”, conforme pode ser verificado na Figura 13.

Figura 13: Cronograma inserido no Autodesk Navisworks

Name	Planned Start	Planned End	Task Type
SERVIÇOS PRELIMINARES	01/04/2019	30/09/2019	Construct
TORRE	05/09/2019	25/06/2020	
JUNTA A	05/09/2019	19/05/2020	
ESTRUTURA DE CONCRETO	05/09/2019	19/05/2020	
Junta A - S51	05/09/2019	19/09/2019	
Junta A - Térreo	20/09/2019	18/10/2019	
Fôrma Térreo A	20/09/2019	04/10/2019	Construct
Desfôrma Térreo A	07/10/2019	18/10/2019	Demolish
Pilares Térreo A	07/10/2019	18/10/2019	Construct
Junta A - 1 pav.	21/10/2019	03/12/2019	
Fôrma 1 PAV A	21/10/2019	14/11/2019	Construct
Desfôrma 1 PAV A	18/11/2019	03/12/2019	Demolish

A próxima etapa para a geração do banco de dados consiste em integrar o modelo BIM com o cronograma inserido na ferramenta *TimeLiner*. Para tanto, serão criados grupos (*sets*) dos elementos BIM presentes no projeto, de forma que cada grupo corresponda a uma etapa específica da obra, de acordo com o cronograma. Foram criados 87 grupos no total. De forma geral, criou-se um grupo (*set*) distinto para cada andar e para cada junta do edifício, separando-se os elementos de pilares, lajes e vigas em conjunto, fôrmas de pilares e bandejas. Um exemplo de grupo com os respectivos elementos pode ser verificado na Figura 14.

Figura 14: Projeto inserido no Autodesk Navisworks, destacando o grupo de fôrmas de pilares do terceiro pavimento



Uma vez feita essa integração entre o modelo e o cronograma, pode-se gerar uma simulação da construção, que pode ser exportada do programa em formato de vídeo ou sequência de imagens. Como o objetivo desse estudo é criar um banco de dados de imagens, escolheu-se a opção de exportação a simulação BIM 4D como uma sequência de imagens da obra. Para tanto, foram geradas 40 simulações, cada uma mostrando o desenvolvimento da obra a partir de um ângulo diferente, escolhido de forma randômica.

Cada simulação gerou um total de 82 imagens. Entretanto, como o objetivo desse estudo é que as imagens geradas pela simulação sejam as mais próximas possíveis das imagens que seriam geradas durante a construção real, foram excluídas do banco de dados as imagens em que há uma diferença significativa entre as atividades que são desenvolvidas no canteiro de obras e a simulação BIM 4D. Após essa filtragem inicial, obteve-se um total de 30 imagens para cada simulação, totalizando um conjunto de 1200 imagens no banco de dados (40 simulações, cada uma com 30 imagens).

Por fim, a última etapa da geração do banco de dados foi integrar cada imagem gerada pela simulação 4D com um progresso percentual da construção dos elementos estruturais presentes na imagem. Uma planilha foi fornecida com o volume de concreto em de cada pilar, viga e laje de cada andar do projeto estrutural. Dessa forma, analisou-se cada imagem gerada e, a partir dos elementos presentes na imagem, obteve-se o volume total de concreto daquela etapa construção representada. Posteriormente, calculou-se o volume total de concreto da construção a partir dos pilares do andar térreo. O progresso percentual da construção dos elementos estruturais em cada imagem foi calculado então segundo a equação abaixo.

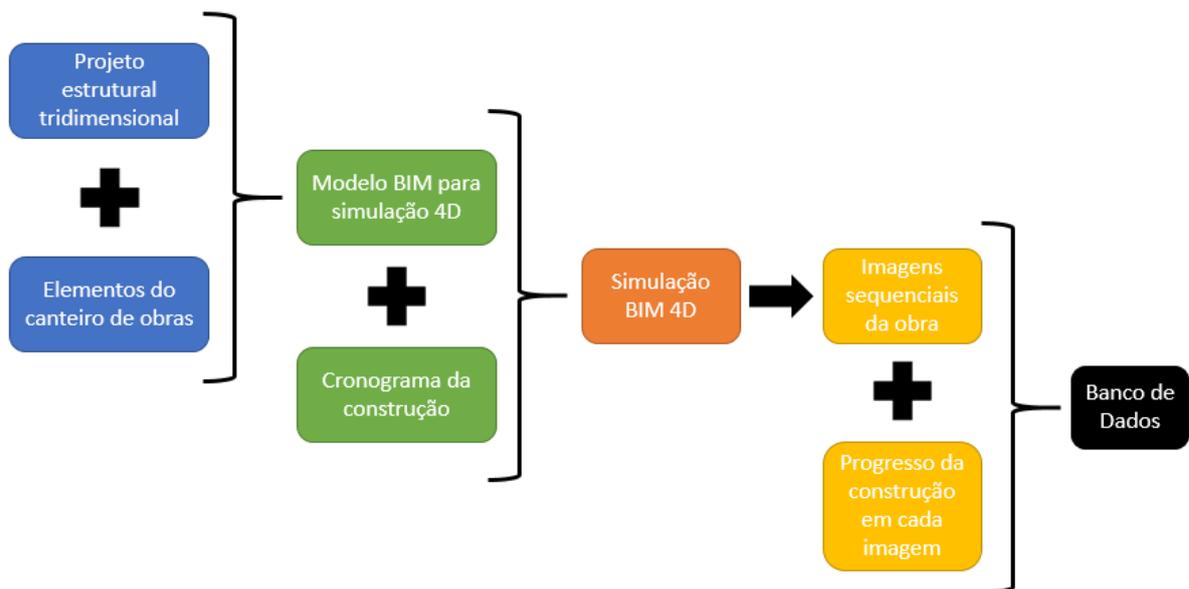
$$\text{Progresso} = \frac{\text{Volume dos elementos estruturais da imagem}}{\text{Volume total dos elementos estruturais da construção}} (\%)$$

Vale destacar que somente foram contabilizados como elemento estrutural construído aqueles elementos que já tinham passado pela etapa da desfôrma na simulação. Dessa forma, caso um elemento já estivesse com a etapa de concretagem concluída, mas ainda não tivesse sido realizada na simulação a etapa de desfôrma, não seria contabilizado aquele volume de concreto para cálculo do progresso da construção dos elementos estruturais.

Uma vez que cada imagem possui o seu respectivo progresso percentual, tem-se que o banco de dados necessário para o desenvolvimento dessa pesquisa está finalizado.

A Figura 15 mostra um fluxograma que esquematiza os procedimentos realizados nessa etapa da pesquisa.

Figura 15: Primeira etapa metodológica da pesquisa



3.2 Parte 2 – Desenvolvimento da estrutura e treinamento das redes neurais convolucionais

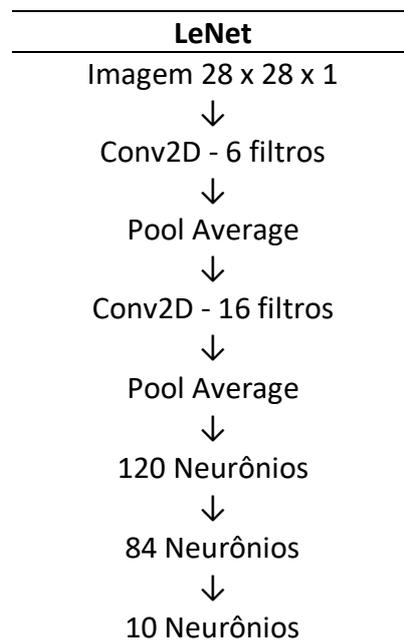
A próxima etapa dessa pesquisa consiste no desenvolvimento de uma estrutura de rede neural convolucional para regressão, bem como no seu treinamento com as imagens do banco de dados gerado na etapa anterior. Optou-se pelo desenvolvimento e

treinamento de 3 estruturas de rede de forma que se possa avaliar de que forma as alterações na estrutura e nos hiperparâmetros das RNCs podem afetar a performance do modelo de aprendizado de máquina.

Uma vez que modelos de RNCs para regressão não são comuns, optou-se tomar por base uma estrutura tradicional de RNC para solução de problemas de classificação e então fazer as alterações necessárias para que a mesma gere regressões, bem como para que possa receber as imagens do banco de dados criado com as imagens do modelo BIM. A estrutura original selecionada foi a LeNet, um dos primeiros modelos de RNC desenvolvido (LeCun *et al.*, 1998), apesar de existirem outras estruturas mais complexas, como a AlexNet.

A LeNet é uma estrutura de rede neural convolucional cujo desenvolvimento se iniciou no ano de 1989, com a proposta de uma estrutura inicial. Em 1998, essa rede foi aprimorada e utilizada em problema de classificação de dígitos escritos à mão. A estrutura da LeNet pode ser verificada na Tabela 1.

Tabela 1: Estrutura da rede neural convolucional LeNet



Como pode se verificar na Tabela 1, a estrutura da LeNet foi feita para receber imagens com o tamanho de 28 pixels de altura e 28 pixels de largura, sendo que a imagem está em uma escala de cinza (apenas um canal de cor), de forma que a estrutura da imagem é 28x28x1, como indicado.

As estruturas responsáveis pelo tratamento da imagem são uma camada de convolução com 6 filtros, seguida por uma camada “*Pool Average*”, seguida por mais uma camada de convolução com 16 filtros e, por fim, uma nova camada “*Pool Average*”. Após o tratamento da imagem, a matriz resultante do tratamento da imagem é submetida a uma operação “*Flatten*”, de forma que a matriz resultante da imagem se transforma em um vetor, que então será submetido à rede neural densa, com uma camada com 120 neurônios e uma camada posterior com 84 neurônios. A última camada da LeNet possui 10 neurônios, que são justificados pelo uso da RNC para classificação de dígitos escritos à mão, o que corresponderiam a 10 valores distintos (algarismos de 0 a 9).

A camada de convolução inicial possui somente 6 filtros, enquanto a segunda possui 16. Isso se deve ao fato de a camada de convolução inicial ser utilizada para captar as principais características da imagem, necessitando, portanto, de uma quantidade menor de filtros, enquanto a segunda camada busca características mais específicas das imagens, o que justifica uma quantidade maior de filtros. As camadas “*Pool Average*”, por sua vez, tem o objetivo de reduzir o tamanho das imagens, conforme destacado anteriormente neste trabalho.

Nesse contexto, o objetivo dessa etapa é, a partir dessa estrutura inicial de RNC, desenvolver outras estruturas de RNC para regressão com imagens de uma simulação BIM 4D. Para tanto, devem ser alterados os dados de entrada da RNC, para receber as imagens da simulação BIM 4D, e a última camada, que deve ser adaptada para realizar uma regressão e não uma classificação. Também serão feitas alterações no número de neurônios e de camadas na rede neural densa, com o objetivo de se captar mais informações das imagens da simulação.

Uma vez que as estruturas forem definidas, o próximo passo é o treinamento das redes neurais convolucionais. Para tanto, o banco de dados criado na Etapa 1 deve ser dividido em dados para treinamento e dados de teste. O objetivo dessa divisão é treinar as redes neurais convolucionais com os dados de treinamento, enquanto a avaliação será feita com os dados de teste, de forma que os algoritmos de aprendizado de máquina sejam avaliados com imagens que não foram utilizadas em seu treinamento, de forma que não haja enviesamento na avaliação do modelo. Tem-se que 75% das imagens (900 imagens) do banco de dados serão utilizados para o treinamento do modelo, enquanto 25% (300 imagens) serão utilizadas para testá-lo.

Tem-se que tanto para a construção das redes neurais convolucionais, treinamento, quanto tratamento do banco de dados de imagens, será utilizada a linguagem de

programação Python, juntamente com diversas bibliotecas próprias para tratamento de dados, manipulação de imagens e aprendizado de máquina, como Pandas, Numpy, Scikit-learn, TensorFlow, Matplotlib e OpenCV. Vale destacar que, para se utilizar a linguagem de programação Python e suas respectivas bibliotecas, será utilizado o ambiente de desenvolvimento integrado Spyder. A seguir, na Tabela 2, será feita uma breve explanação sobre o Python e as bibliotecas utilizadas nessa etapa da pesquisa.

Tabela 2: Ferramentas de programação utilizadas nessa pesquisa

Ferramenta	Descrição
Python	O Python é uma linguagem de programação com diversas aplicações, entre elas desenvolvimento Web, computação científica e numérica e desenvolvimento de <i>softwares</i> (Python, 2023).
Pandas	O Pandas, por sua vez, é uma biblioteca construída com a linguagem de programação Python. Entre as suas diversas funcionalidades estão a criação de <i>DataFrames</i> , além da leitura e manipulação de arquivos CSV, do MS Excel e SQL. O Pandas também possui uma série de recursos para trabalhar com manipulação de dados, como tratamento de dados faltantes, combinação de conjuntos de dados, entre outros (Pandas, 2023).
NumPy	O NumPy também é uma biblioteca da linguagem de programação Python, utilizada para computação científica. Entre as diversas funcionalidades dessa biblioteca, está a capacidade de trabalhar com matriz de n dimensões (NumPy, 2023).
TensorFlow	O TensorFlow é uma plataforma de código aberto para criação e desenvolvimento de algoritmos de aprendizado de máquina. O TensorFlow opera tanto na área de preparação dos dados, como na criação e implantação de modelos de aprendizado de máquina (TensorFlow, 2023).
OpenCV	O OpenCV (Open Source Computer Vision Library) é uma plataforma de código aberto utilizada na área da visão computacional e aprendizado de máquina, possuindo mais de 2500 algoritmos otimizados (OpenCV, 2023).

Scikit-learn	O scikit-learn é uma plataforma de aplicações de aprendizado de máquina em Python. Entre as diversas funcionalidades dessa biblioteca, estão algoritmos para classificação, regressão e agrupamento, além de ferramentas de pré-processamento de dados e seleção de modelos (scikit-learn, 2023).
Matplotlib	A biblioteca Matplotlib, uma biblioteca da linguagem de programação Python utilizada para criar visualizações estáticas, dinâmicas e interativas de dados (Matplotlib, 2023).

Superada essa etapa, o próximo passo será o efetivo treinamento das redes neurais convolucionais. Se for notado durante essa etapa que os resultados da rede não estão evoluindo, serão feitas modificações pontuais nos hiperparâmetros da mesma até que se possa verificar que as RNCs estão melhorando sua performance. Após o treinamento das redes, os resultados serão armazenados em um arquivo de extensão “h5”, que é a extensão para arquivos “Hierarchical Data Format”, de forma que as RNCs possam ser utilizadas posteriormente, sem necessidade de se realizar um novo treinamento.

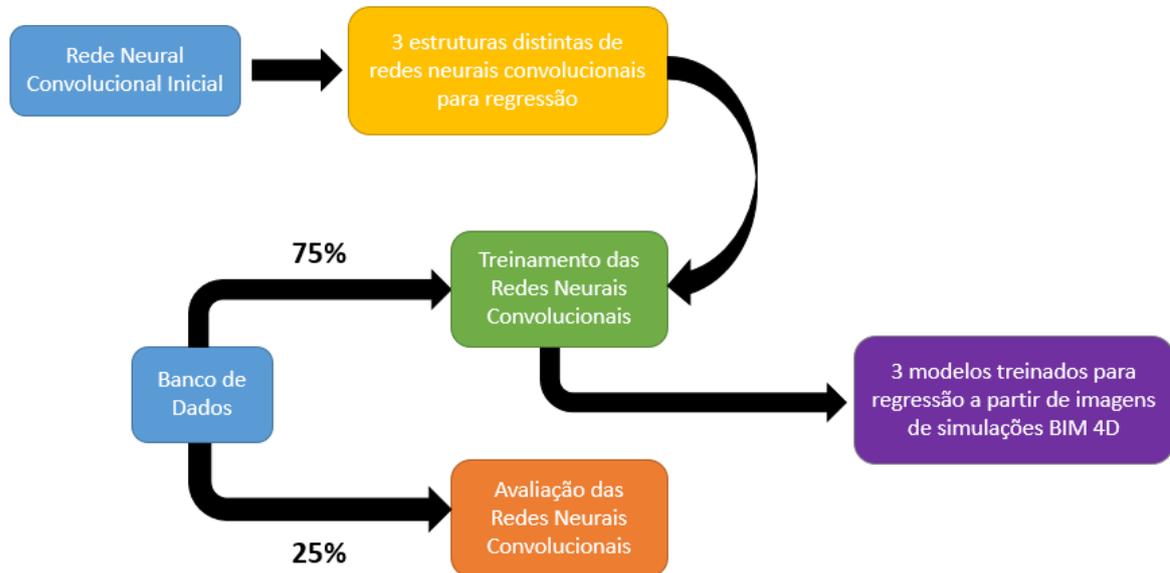
O produto da Parte 2 dessa pesquisa, portanto, será um conjunto de 3 redes neurais convolucionais treinadas com imagens provenientes da simulação de uma obra. A Tabela 3 mostra as bibliotecas do Python utilizadas para cada tarefa dessa etapa.

Tabela 3: Ferramentas utilizadas em cada etapa da pesquisa

Tarefa	Biblioteca do Python utilizada
Manipulação e impressão de imagens	OpenCV, Matplotlib, Pandas, Numpy
Criação e manipulação de DataFrames	Pandas, Numpy, Scikit-learn
Criação e treinamento das redes neurais convolucionais	Tensorflow
Avaliação estatística dos modelos	Scikit-learn

A Figura 16 mostra de forma esquemática um fluxograma dessa etapa da pesquisa.

Figura 16: Segunda etapa metodológica da pesquisa



3.3 Parte 3 - Avaliação estatística das redes neurais convolucionais geradas

A Parte 3 dessa pesquisa consiste em uma avaliação estatística dos resultados gerados pelas redes neurais convolucionais treinadas. O objetivo é verificar a performance das redes, se os resultados são confiáveis, bem como da ocorrência de *overfitting* ou *underfitting*.

O primeiro passo dessa etapa é submeter as imagens separadas para teste da rede neural às 3 RNCs, de forma que as mesmas gerem previsões do progresso da obra em cada uma das imagens. Uma vez que todas as previsões tiverem sido realizadas, elas serão comparadas com os resultados originais, segundo métricas estatísticas.

A biblioteca do Python scikit-learn oferece uma série de opções para avaliação do resultado de regressões de modelos de aprendizado de máquina, como por exemplo o *Mean Absolute Error*, o *Root Mean Squared Error*, o *Root Mean Squared Logarithmic Error*, entre outros. Nessa pesquisa, os resultados serão analisados segundo duas métricas de regressão.

A primeira avaliação será por meio do Erro Médio Absoluto (*Mean Absolute Error*- MAE), que pode ser obtido segundo a fórmula abaixo.

$$MAE = \frac{\sum_{i=1}^n |Valor\ previsto_i - Valor\ Real_i|}{n}$$

O objetivo do MAE é calcular, em média, o quanto os valores previstos das redes neurais convolucionais estão distantes dos valores reais, de forma que se possa verificar a confiabilidade dessas previsões.

A segunda avaliação será feita pelo cálculo da Raiz Quadrática Média (*Root Mean Squared Error*), que pode ser obtido pela fórmula abaixo.

$$RMSE = \sqrt{\sum_{i=1}^n \frac{(Valor\ previsto_i - Valor\ Real_i)^2}{n}}$$

A escolha dessas duas métricas é devido ao fato de que, segundo KARUNASINGHA (2022), elas são duas das principais medidas de erros utilizadas em diversas áreas do conhecimento, entre elas a mineração de dados e o aprendizado de máquina (TANG *et al.*, 2017).

Uma vez que as imagens de teste forem avaliadas, as imagens separadas para o treinamento das redes neurais convolucionais também serão submetidas às RNCs treinadas e às avaliações estatísticas, com o objetivo de se avaliar a ocorrência de *underfitting* ou de *overfitting* no modelo de aprendizado de máquina.

O *underfitting* ocorre quando um modelo de aprendizado de máquina é demasiado simples para o problema que pretende solucionar, de forma que não consegue captar as principais características dos dados fornecidos para seu treinamento. Dessa maneira, tem-se que os resultados do modelo tanto nos dados de treinamento quanto nos dados de teste são ruins (Everitt, 2010).

O *overfitting*, por outro lado, ocorre quando um modelo complexo consegue se adaptar muito aos dados de treinamento, de forma que deixa de captar padrões gerais nos dados e ocorre uma espécie de “memorização” desses dados. Dessa maneira, tem-se que o modelo apresenta uma alta precisão quando aplicado aos dados com que foi treinado, mas uma performance ruim em outros dados, como os de teste (Cárdenas-Montes, 2006).

De forma, resumida, o *overfitting* e o *underfitting* podem ser verificados conforme o resumo da Tabela 4.

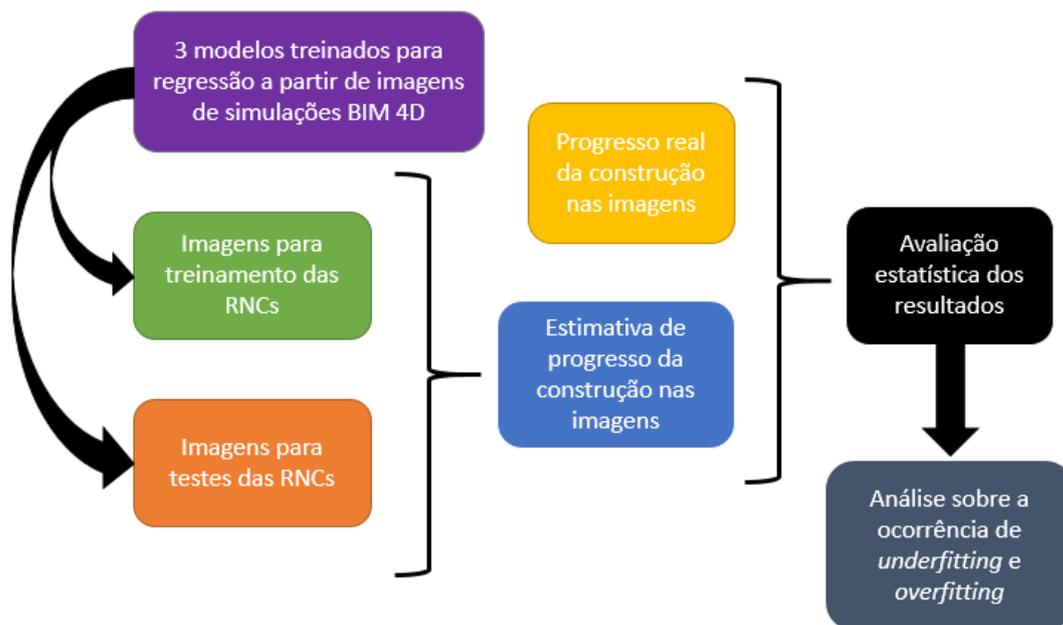
Tabela 4: Avaliação do modelo de aprendizado de máquina

Performance nos Dados de Treinamento	Performance nos Dados de Teste	Avaliação do Modelo
Bom	Bom	Modelo adequado
Bom	Ruim	<i>Overfitting</i>
Ruim	Ruim	<i>Underfitting</i>

Para se realizar a avaliação estatística dos resultados das redes neurais convolucionais, será utilizada novamente a linguagem de programação Python, mais especificamente a biblioteca Scikit-learn.

O resultado dessa Parte da pesquisa, portanto, é uma avaliação das RNCs criadas, de forma que se possa verificar de forma estruturada a sua confiabilidade. Um esquema dessa etapa do trabalho pode ser verificado na Figura 17.

Figura 17: Terceira etapa metodológica da pesquisa



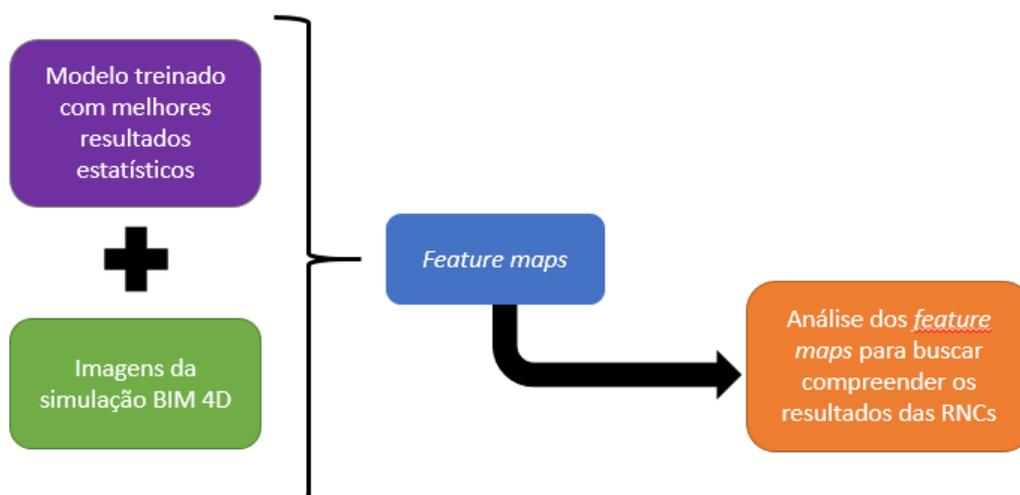
3.4 Parte 4 - Análise dos *feature maps* gerados pelas redes neurais convolucionais

A próxima etapa dessa pesquisa será, a partir das imagens da simulação BIM 4D e dos modelos de redes neurais convolucionais treinados, avaliar os *feature maps* gerados quando as imagens são submetidas ao modelo de aprendizado de máquina. O objetivo desta etapa é buscar identificar quais foram as características das imagens que as RNCs efetivamente identificaram como relevantes e úteis para estimar o progresso da construção.

Para tanto, serão selecionadas 3 imagens aleatórias da simulação BIM 4D e elas serão submetidas à rede neural convolucional que apresentou os melhores resultados estatísticos, ou seja, MAE e RMSE mais baixos. Serão então gerados os mapas de características após as imagens da simulação serem submetidas à primeira camada de convolução, à primeira camada de pooling e à segunda camada de convolução. Os *feature maps* serão então avaliados verificando-se quais foram os elementos destacados nas imagens, bem como a posição relativa entre esses elementos.

O objetivo desta etapa é busca explicar os resultados das RNCs, de forma a se ter uma melhor compreensão de como o modelo treinado chega aos resultados de progresso da construção a partir das imagens da simulação BIM 4D. Um esquema do desenvolvimento desta etapa pode ser verificado na Figura 18.

Figura 18: Quarta etapa metodológica da pesquisa



3.5 Visão geral do desenvolvimento do trabalho

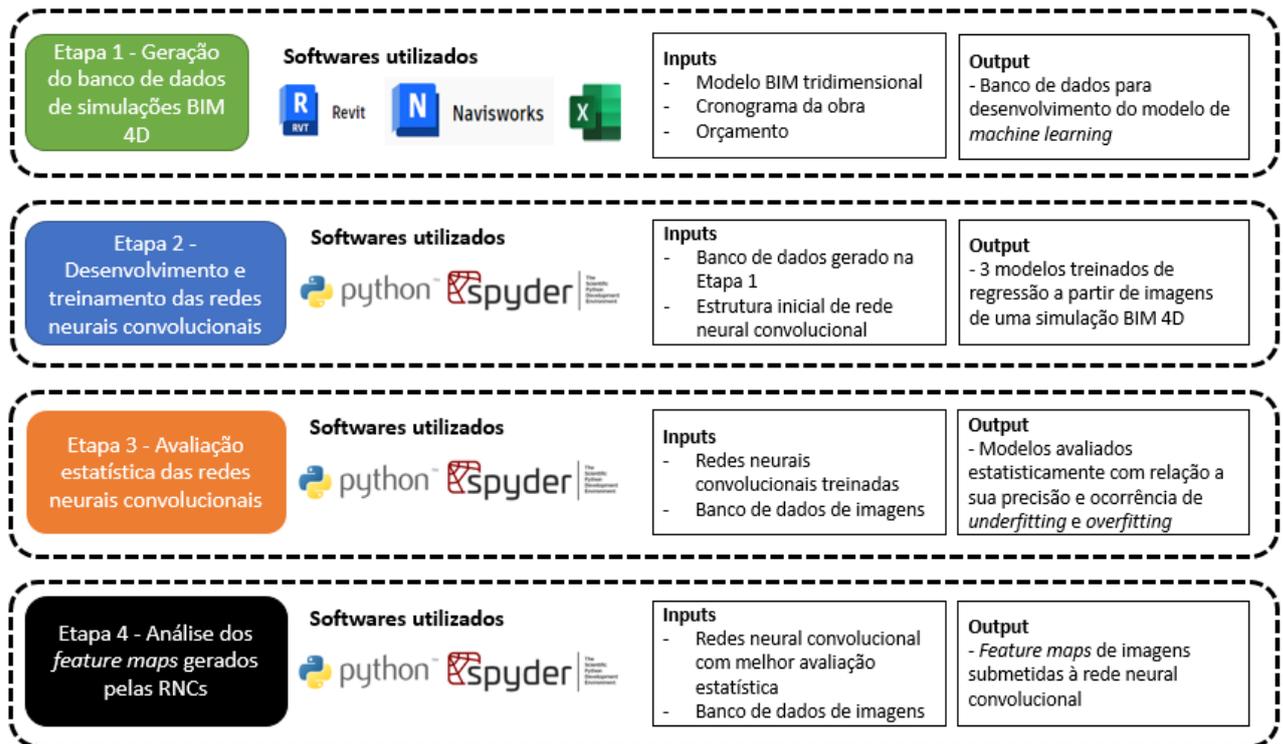
Uma visão geral de cada uma das etapas deste trabalho, com os respectivos softwares e ferramentas utilizadas em cada etapa pode ser verificada na Tabela 5 e na Figura 19.

Tabela 5: Etapas da pesquisa

Etapa	Softwares e ferramentas utilizadas
Parte 1 – Desenvolvimento de um banco de dados de imagens de uma simulação BIM 4D com o respectivo progresso da construção	- Autodesk Revit - Autodesk Navisworks - MS Project - MS Excel
Parte 2 – Desenvolvimento da estrutura e treinamento das redes neurais convolucionais	- Linguagem de programação Python com o uso das bibliotecas Pandas, Numpy, OpenCV, Sklearn, Matplotlib e TensorFlow
Parte 3 - Avaliação estatística das redes neurais convolucionais geradas	- Linguagem de programação Python com o uso das bibliotecas Pandas, Numpy, OpenCV, Sklearn, Matplotlib e TensorFlow
Parte 4 - Análise dos <i>feature maps</i> gerados pelas redes neurais convolucionais	- Linguagem de programação Python com o uso das bibliotecas Pandas, Numpy, OpenCV, Sklearn, Matplotlib e TensorFlow

Fonte: do autor

Figura 19: Esquema das etapas metodológicas da pesquisa



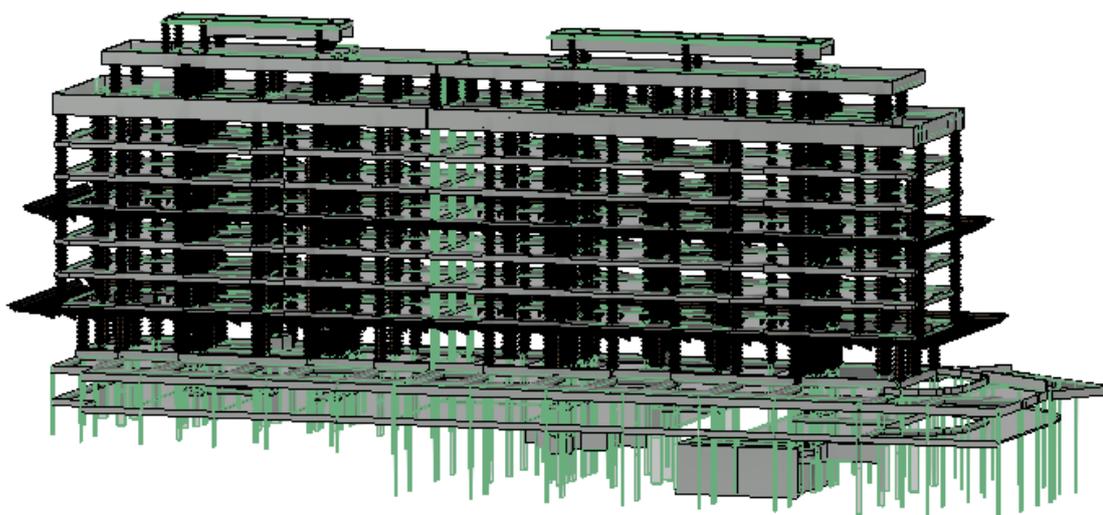
Fonte: do autor

4 RESULTADOS E DISCUSSÃO

4.1 Parte 1 – Desenvolvimento de um banco de dados de imagens de uma simulação BIM 4D com o respectivo progresso da construção

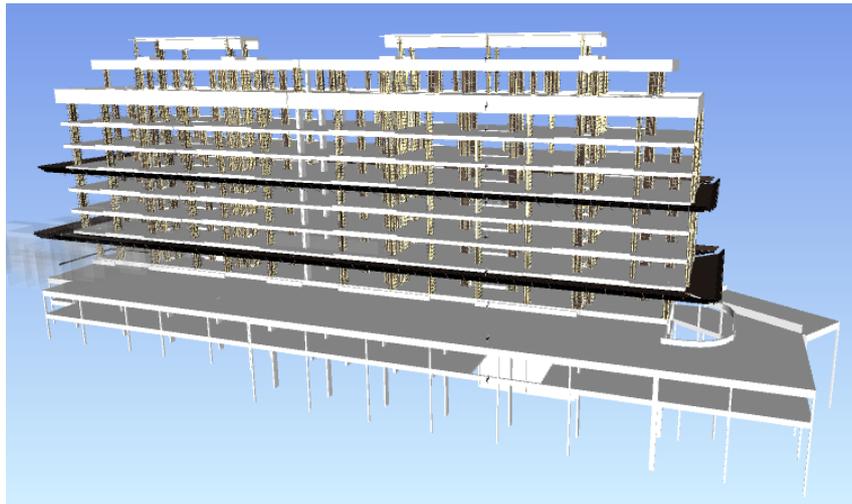
Como destacado anteriormente na metodologia da pesquisa, a primeira etapa desse trabalho consistiu na criação do banco de dados com imagens de uma simulação da construção da parte estrutural de um edifício de 6 pavimentos. Para tanto, tomou-se como base o projeto estrutural em formato IFC inserido dentro do software Autodesk Revit e modelou-se as fôrmas dos pilares e bandejas. A Figura 20 mostra o resultado dessa modelagem.

Figura 20: Modelo tridimensional estrutural com elementos temporários da construção



Esses modelos tridimensionais foram inseridos no software Autodesk Navisworks, conforme pode ser verificado na Figura 21, e seus elementos foram segregados em grupos que fossem compatíveis com as tarefas do cronograma da construção, também inserido no *software*. Uma vez que os elementos do modelo BIM e as tarefas do cronograma estavam conectados, foi possível gerar uma simulação de algumas etapas da construção. Uma vez que a simulação foi efetuada, o software fornece a opção para que ela seja exportada em formato de vídeo ou em sequência de imagens. Como um dos objetivos dessa pesquisa é treinar uma rede neural convolucional, optou-se por exportar as simulações geradas em formato de imagens JPG.

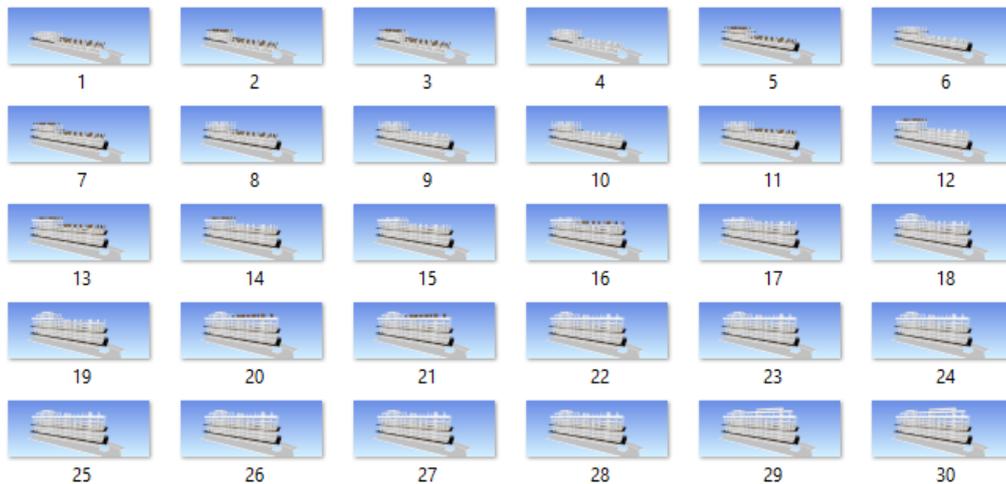
Figura 21: Modelo BIM no software Autodesk Navisworks.



Fonte: Do autor

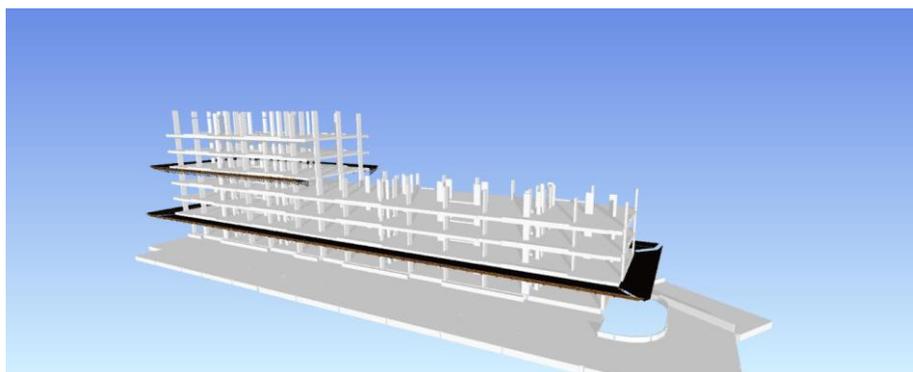
Com o intuito de se criar um banco de dados robusto, foram feitas simulações da obra de vários ângulos distintos, selecionados de forma manual. No total, foram geradas 40 simulações. Um exemplo de simulação pode ser verificado na Figura 22 e uma imagem de uma etapa específica da simulação na Figura 23.

Figura 22: Exemplo de simulação gerada no Autodesk Navisworks



Fonte: Do autor

Figura 23: Exemplo de imagem gerada na simulação



Fonte: Do autor

A próxima etapa foi a exclusão das imagens geradas que não agregariam ao banco de dados, como por exemplo imagens sem nenhum elemento visível e que representavam etapas iniciais da obra, não representando de forma adequada a real situação do canteiro naquele momento.

Após a eliminação dessas imagens, obteve-se um total de 30 imagens por simulação, totalizando 1200 imagens, que mostravam uma simulação da obra por diversos ângulos distintos. Todas as simulações com todas as respectivas imagens geradas podem ser encontradas no Apêndice I desse trabalho.

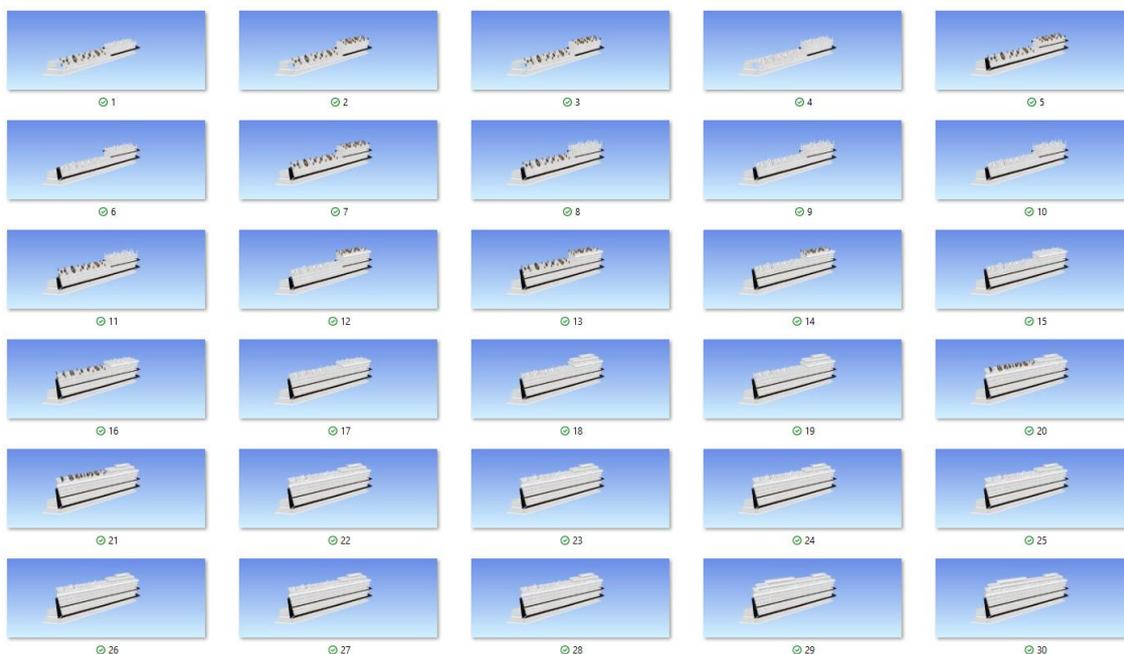
Para facilitar a manipulação de todas as imagens geradas para o banco de dados dentro da linguagem de programação Python, utilizada nas próximas etapas dessa pesquisa, foi necessário renomear os arquivos, de forma que pudessem ser facilmente identificados e estivessem em uma ordem lógica. Dessa forma, cada imagem de cada uma das 30 simulações foi identificada com um número de 1 a 40, conforme a ordem que apareciam na simulação. O código em Python para renomear as 1200 imagens de forma automatizada pode ser verificado no Apêndice II desta pesquisa.

A próxima etapa foi a avaliação das imagens geradas e identificar os elementos da construção que aparecem em cada uma. De posse do orçamento da obra, foi possível calcular o volume de concreto presente em cada imagem. Dividindo-se a quantidade total de concreto em cada imagem pelo volume total de concreto da construção, obteve-se o percentual da obra concluído até aquele momento. Com a conexão entre as imagens da simulação e o respectivo progresso da obra em cada uma das imagens, o banco de dados para treinamento e teste das redes neurais convolucionais estava completo. Na Tabela 6, observa-se o progresso calculado da construção de elementos estruturais para cada imagem da simulação, com a Figura 24 de exemplo de simulação.

Tabela 6: Progresso calculado para cada imagem de cada simulação

Progresso calculado para cada imagem da simulação					
Imagem	Progresso calculado	Imagem	Progresso calculado	Imagem	Progresso calculado
Imagem 1	0,41	Imagem 11	0,70	Imagem 21	0,96
Imagem 2	0,45	Imagem 12	0,76	Imagem 22	0,96
Imagem 3	0,45	Imagem 13	0,81	Imagem 23	0,96
Imagem 4	0,52	Imagem 14	0,81	Imagem 24	0,96
Imagem 5	0,56	Imagem 15	0,82	Imagem 25	0,96
Imagem 6	0,60	Imagem 16	0,86	Imagem 26	0,96
Imagem 7	0,64	Imagem 17	0,87	Imagem 27	0,96
Imagem 8	0,65	Imagem 18	0,90	Imagem 28	0,96
Imagem 9	0,66	Imagem 19	0,90	Imagem 29	1,00
Imagem 10	0,66	Imagem 20	0,96	Imagem 30	1,00

Figura 24: Exemplo de simulação com imagens sequenciais da construção



Para que o banco de dados pudesse ser utilizado na pesquisa para o desenvolvimento de um modelo de aprendizado de máquina, o mesmo teve de ser inserido

no ambiente da linguagem de programação Python. Para tanto, foram utilizadas as bibliotecas NumPy e Pandas para armazenar esses dados, mais especificamente em um objeto do tipo Array do NumPy e DataFrame do Pandas.

Uma vez que os dados estavam inseridos no ambiente Python, o próximo passo seria dividir os dados em treinamento e teste, em que os dados de treinamento serão utilizados para o treinamento da rede neural convolucional e os dados de teste para a avaliação das redes treinadas. Para realizar essa divisão, utilizou-se uma biblioteca do Python denominada Scikit-learn, que possui a função “train_test_split”, que gera como resultado uma divisão de banco de dados, sendo uma parte para treinamento e outra para teste das redes neurais. Nessa pesquisa, optou-se por utilizar 25% dos dados para avaliação do modelo e 75% para treinamento, de forma que 900 imagens serão utilizadas para treinamento do algoritmo de aprendizado de máquina e 300 para o teste desse algoritmo gerado. Também foi inserido nessa função um parâmetro denominado “random_state”, que possibilita que a mesma divisão seja gerada posteriormente, caso se deseje repetir a pesquisa com os exatos dados de treinamento e teste.

O código utilizado para a inserção do banco de dados no ambiente de programação, bem como o código para divisão do banco de dados em treinamento e teste pode ser encontrado no Apêndice III desse trabalho.

4.2 Parte 2 – Desenvolvimento da estrutura e treinamento das redes neurais convolucionais

A próxima etapa dessa pesquisa foi o desenvolvimento da estrutura das redes neurais convolucionais e seu respectivo treinamento. Para desenvolvimento das novas estruturas, tomou-se por base a estrutura de rede LeNet, como apresentado anteriormente.

A primeira alteração na estrutura da rede foi a modificação do tamanho das imagens que seriam os dados de entrada para rede. Na LeNet original, as imagens eram em escala de cinza possuindo um tamanho de 28x28, sendo utilizada para classificação de dígitos escritos à mão. As imagens geradas nessa pesquisa são coloridas, possuindo, portanto, os 3 canais de cores (vermelho, verde e azul). Além disso, por serem imagens com mais detalhes do que dígitos escritos à mão, optou-se por aumentar o tamanho dos dados de entrada das redes para um tamanho de 256x256, de forma que menos informações da simulação da construção sejam perdidas quando se reduzir o tamanho das

imagens. Dessa forma, enquanto na rede LeNet inicial a entrada da rede era uma imagem 28x28x1, nessa pesquisa os dados de entrada são do formato 256x256x3.

Não foram feitas alterações nas camadas de convolução na estrutura da rede neural convolucional, de forma que foram mantidas as duas camadas de convolução, sendo a primeira com 6 filtros e a segunda com 16, correspondendo à primeira e terceira camada da RNC, respectivamente.

A camada de “*Pooling*” da RNC também foi alterada, de forma que a camada original de *Average Pool* foi substituída por uma camada do tipo *Max Pool*, mais utilizada em redes neurais convolucionais modernas, como por exemplo a rede AlexNet. A diferença entre as duas camadas é que, enquanto a do tipo *Pool Average* reduz a imagem de entrada da RNC tomando uma média dos valores de uma determinada região da imagem, a *Max Pool* reduz o tamanho da imagem tomando o maior valor da mesma região. Feitas essas alterações, tem-se que respectivamente a segunda e quarta camada das RNCs são camadas do tipo *Max Pool*.

A próxima camada da RNC se manteve semelhante à rede LeNet original, se tratando de uma camada do tipo *Flatten*, que simplesmente transforma a imagem em um vetor, que poderá então servir de entrada para a rede neural densa que efetivamente fará a regressão. A diferença entre as 3 redes neurais convolucionais geradas começa a partir da camada *Flatten*, ou seja, a estrutura somente se difere na parte referente à rede neural densa.

A rede neural convolucional 1 (RNC_1) tem como diferença em relação à LeNet a quantidade de camadas e o respectivo número de neurônios em cada camada. Enquanto a LeNet possuía uma camada com 120 neurônios seguida por uma com 84 neurônios, a RNC 1 possui uma única camada com 128 neurônios. A estrutura da rede neural densa das redes neurais 2 e 3 (RNC_2 e RNC_3, respectivamente) foram baseadas na estrutura da RNC_1. Na RNC_2, adicionou-se mais uma camada com 128 neurônios, enquanto na RNC_3 a camada existente passou a ter 256 neurônios. O objetivo dessas modificações foi avaliar o efeito do aumento do número de camadas e do aumento do número de neurônios em uma camada para os resultados da RNC.

Por fim, os 10 neurônios finais da LeNet foram substituídos por um único neurônio para realização da regressão, com uma função de ativação “*Rectified Linear Unit*” (ReLU).

A estrutura das redes neurais convolucionais criadas pode ser verificada na Tabela 7. A Tabela 8, por sua vez, mostra a quantidade de parâmetros treináveis em cada rede neural convolucional.

Tabela 7: Estrutura das redes neurais convolucionais

RNC_1	RNC_2	RNC_3
Imagem 256 x 256 x 3	Imagem 256 x 256 x 3	Imagem 256 x 256 x 3
↓	↓	↓
Conv2D - 6 filtros	Conv2D - 6 filtros	Conv2D - 6 filtros
↓	↓	↓
MaxPool2D	MaxPool2D	MaxPool2D
↓	↓	↓
Conv2D - 16 filtros	Conv2D - 16 filtros	Conv2D - 16 filtros
↓	↓	↓
MaxPool2D	MaxPool2D	MaxPool2D
↓	↓	↓
128 Neurônios	128 Neurônios	256 Neurônios
↓	↓	↓
1 Neurônio	128 Neurônios	1 Neurônio
	↓	
	1 Neurônio	

Tabela 8: Quantidade de parâmetros treináveis em cada RNC treinada

Rede Neural Convolucional	Parâmetros Treináveis
RNC 1	1.387.577
RNC 2	1.404.089
RNC 3	2.987.833

Como se pode verificar na Tabela 8, a RNC 1 é a estrutura gerada com o menor número de parâmetros treináveis. Em seguida, está a RNC 2, de forma que foi possível notar que o acréscimo de uma camada com 128 neurônios aumentou o número de parâmetros treináveis em 16.512. Por fim, a RNC 3 é a rede com o maior número de parâmetros treináveis, com mais que o dobro do valor das RNC 1 e 2. Dessa forma, avaliando-se os resultados das 3 RNCs, pode-se verificar se o aumento do número de neurônios em uma rede acaba por corresponder em uma melhora da sua performance.

O site <netron.app> permite que seja gerado um fluxograma da estrutura de uma RNC, de forma que se possa visualizar as camadas que compõem uma rede, sua estrutura, bem como a relação entre as mesmas. O aplicativo foi utilizado para gerar a estrutura das

3 redes neurais convolucionais desse trabalho, como pode ser verificado no Apêndice IV desta pesquisa.

Uma vez que as estruturas das RNCs foram criadas, utilizou-se a biblioteca TensorFlow para gerar essas estruturas no ambiente do Python e treiná-las com as imagens do banco de dados separadas para treinamento. O treinamento ocorreu através do comando *fit*. Em seguida, os resultados do treinamento das redes foram salvos em um arquivo Hierarchical Data Format, de extensão “h5”. Por fim, as redes foram novamente carregadas no ambiente Python, com o uso do comando *load*.

O código em Python referente a essa etapa pode ser encontrado no Apêndice V deste trabalho.

4.3 Parte 3 - Avaliação estatística das redes neurais convolucionais geradas

Uma vez que as redes neurais convolucionais foram treinadas, o próximo passo é avaliar sua performance nos dados de teste, de forma a verificar sua precisão quando utilizada para avaliar dados com as quais a rede não teve contato anterior. Além disso, os dados utilizados para o treinamento das RNCs também serão submetidos à rede, de forma que os resultados combinados sobre o desempenho do modelo nos dados de treinamento e teste possam indicar a ocorrência de *underfitting* e *overfitting* nas redes neurais convolucionais treinadas.

Como forma de exemplificar a precisão e o funcionamento dos modelos treinados, serão utilizadas as 30 imagens de uma simulação (Figura 25), que serão submetidas à primeira rede neural treinada (RNC_1), de forma que a rede possa prever o progresso da construção representado em cada figura, comparado com o progresso previamente calculado por meio das tabelas de quantitativos da obra. A Tabela 9 faz a comparação entre os resultados da rede neural e os progressos originais calculados.

Figura 25: Exemplo de simulação BIM 4D

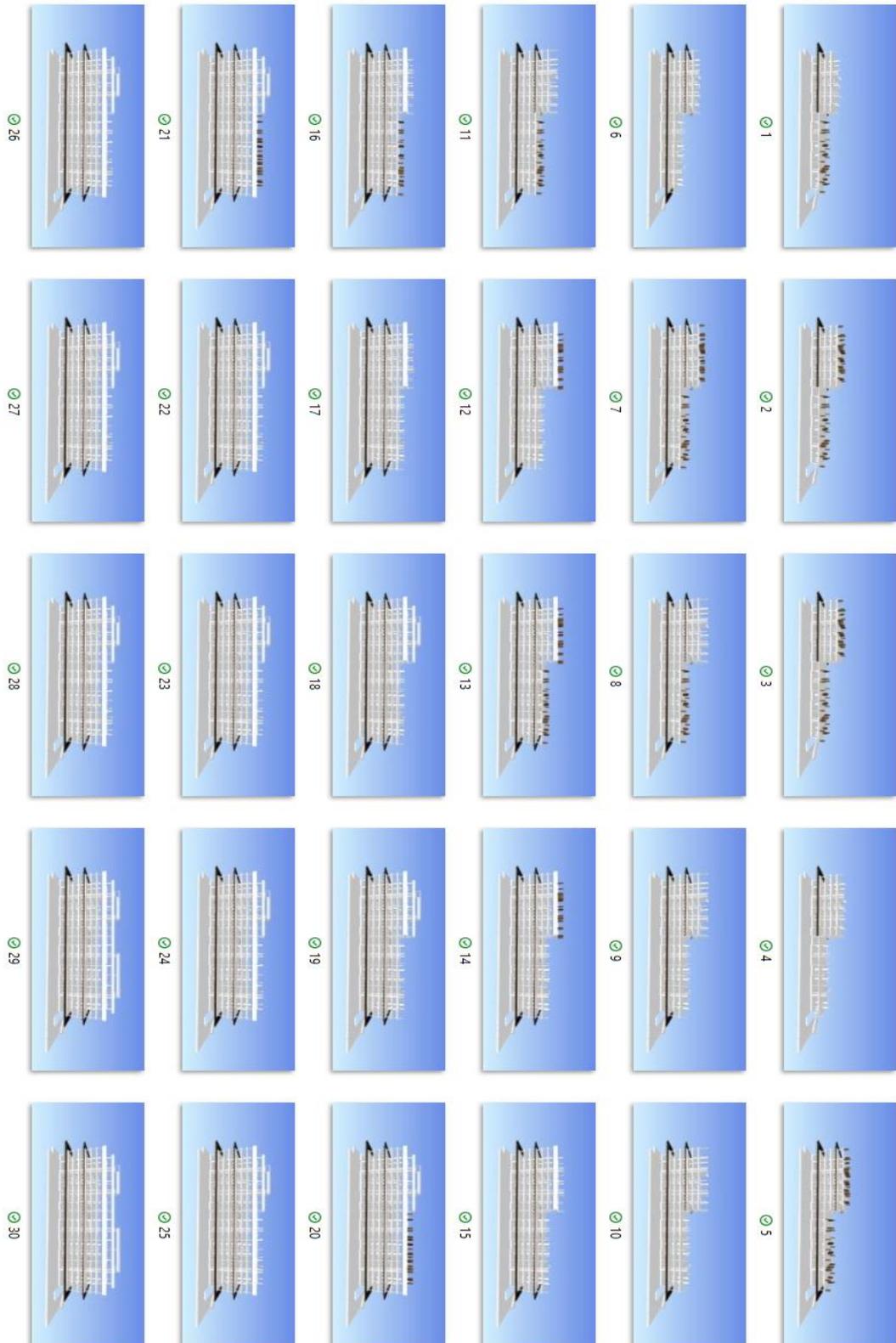


Tabela 9: Comparação do progresso real e estimado para a simulação exemplo

Imagem	Progresso real	Progresso estimado	Diferença (Módulo)
Imagem 1	41,39%	42,09%	0,71%
Imagem 2	45,03%	45,19%	0,17%
Imagem 3	45,03%	45,19%	0,17%
Imagem 4	51,71%	44,72%	6,99%
Imagem 5	55,92%	57,38%	1,46%
Imagem 6	60,08%	59,19%	0,89%
Imagem 7	64,29%	63,39%	0,90%
Imagem 8	64,94%	64,24%	0,71%
Imagem 9	65,64%	65,19%	0,45%
Imagem 10	65,64%	65,19%	0,45%
Imagem 11	70,37%	68,32%	2,05%
Imagem 12	75,97%	74,93%	1,04%
Imagem 13	80,70%	79,41%	1,29%
Imagem 14	81,40%	81,03%	0,36%
Imagem 15	81,66%	81,97%	0,31%
Imagem 16	86,39%	85,09%	1,30%
Imagem 17	87,12%	85,70%	1,43%
Imagem 18	89,62%	89,07%	0,55%
Imagem 19	89,62%	89,07%	0,55%
Imagem 20	96,07%	94,65%	1,42%
Imagem 21	96,07%	94,65%	1,42%
Imagem 22	96,40%	95,31%	1,09%
Imagem 23	96,40%	95,31%	1,09%
Imagem 24	96,40%	95,31%	1,09%
Imagem 25	96,40%	95,31%	1,09%
Imagem 26	96,40%	95,31%	1,09%
Imagem 27	96,40%	95,31%	1,09%
Imagem 28	96,40%	95,31%	1,09%

Imagem 29	100,00%	99,58%	0,42%
Imagem 30	100,00%	99,58%	0,42%

A análise da performance das RNCs será feita utilizando duas métricas, como explicitado anteriormente: o Erro Médio Absoluto (EMA) e o *Root Mean Squared Error* (RMSE). Ao submeter os dados de treinamento e teste às RNCs 1,2 e 3, obteve-se os resultados explicitados na Tabela 10.

Tabela 10: Avaliação estatística das redes neurais convolucionais

Rede Neural Convolucional	Precisão nos dados de treinamento		Precisão nos dados de teste	
	MAE	RMSE	MAE	RMSE
RNC_1	0,82%	1,04%	1,93%	2,90%
RNC_2	1,21%	1,51%	2,29%	3,15%
RNC_3	2,28%	2,73%	3,10%	3,77%

Ao analisar os resultados obtidos pelas redes neurais convolucionais nos dados de treinamento, pode-se concluir que não houve *underfitting* no modelo, uma vez que a precisão do mesmo nos dados utilizados para treinar as redes neurais é alta, com o MAE variando de 0,82% a 2,28%, indicando que o modelo é adaptável aos dados fornecidos, captando suas principais características e fornecendo resultados adequados.

A análise do modelo nos dados de teste também permite concluir que não houve *overfitting*, uma vez que também há uma alta precisão na previsão realizada, com o MAE variando de 1,93% a 3,10%, indicando que o modelo performa bem com dados não utilizados em seu treinamento, não havendo, portanto, uma “memorização” das características das imagens utilizadas no treino das RNCs, e sim uma captação de seus atributos gerais e o entendimento de um padrão que pode ser replicado em imagens inéditas para os modelos já treinados.

Uma vez que não houve *overfitting* nem *underfitting*, pode-se concluir que os modelos de redes neurais convolucionais gerados são adequados e funcionais, conseguindo captar as principais características das imagens e vinculá-las ao respectivo progresso da obra representado. Depreende-se, portanto, a possibilidade de se integrar o

BIM com as redes neurais convolucionais para a solução de problemas de regressão, um dos objetivos primordiais desta pesquisa.

Posteriormente, deve-se analisar a performance individual de cada rede neural treinada. Avaliando os resultados da Tabela 10, tem-se que a RNC 1 foi a que apresentou os melhores resultados, seguida pela RNC 2 e, por último, a RNC 3. Dessa forma, pode-se verificar que, apesar da RNC 3 ter o maior número de parâmetros treináveis, ela não obteve a maior precisão entre os modelos, enquanto a RNC 1, que possuía o menor número de parâmetros treináveis, apresentou os resultados mais precisos. Dessa forma, pode-se concluir que, para esses modelos específicos de aprendizado de máquina, combinados com os dados fornecidos para treino, o acréscimo de uma camada na rede neural ou o aumento do número de neurônios em uma camada não resulta em um aumento da precisão da rede neural, ou seja, a complexidade do modelo não está diretamente relacionada, neste caso, com a precisão dos seus resultados.

Segundo Uzair e Jamil (2020), de forma geral, quando se aumenta a complexidade de uma rede neural, como por exemplo com o acréscimo do número de camadas ocultas, ocorre uma melhora na performance do modelo, apesar de se necessitar de um tempo maior de treinamento do mesmo. Todavia, destaca-se que, ao se aumentar desnecessariamente a complexidade do modelo, com acréscimo de camadas e da quantidade de neurônios em cada camada, pode ocorrer o problema de *overfitting*, ou seja, o modelo, por ser demasiado complexo, acaba por captar nuances dos dados de treinamento e perde sua capacidade de generalização.

De acordo com Heaton *et al.* (2018), a generalização de um modelo pode ser entendida como a sua capacidade de captar as características gerais dos dados, apresentando uma performance adequada em dados não utilizados em seu treinamento. Reed e Marksii (1999), por sua vez, destacaram que a capacidade de generalização dos algoritmos possui um limite, de forma a se chegar a uma precisão máxima ou erro mínimo. Ao se continuar aumentando a complexidade do modelo, entretanto, o algoritmo começa a captar especificidades dos dados e perde sua capacidade de generalização, de forma que o erro geral do modelo aumenta, podendo ocorrer inclusive o fenômeno de *overfitting*.

No caso dessa pesquisa, tem-se que, devido ao baixo MAE e RMSE em todas as redes neurais convolucionais treinadas, não houve a ocorrência de *overfitting*. Entretanto, nota-se que as redes RNC_2 e RNC_3, que possuíam o maior número de parâmetros treináveis, apresentaram resultados inferiores ao da RNC_1, uma rede considerada mais

simples. Dessa forma, pode-se concluir que, apesar das redes RNC_2 e RNC_3 ainda serem algoritmos válidos e com bons resultados, não houve aumento de performance do modelo com o aumento do número de parâmetros.

Outro motivo que pode explicar o menor rendimento das redes RNC_2 e RNC_3 é o fato de que muitos parâmetros no treinamento das RNCs iniciam com valores aleatórios, de forma que essa aleatoriedade pode favorecer um modelo em detrimento de outro. As imagens selecionadas para treinamento e os pesos iniciais das redes neurais convolucionais, por exemplo, foram selecionados de forma randômica, sendo que a combinação desses fatores pode ter desfavorecido as RNC_2 e RNC_3.

Apesar dos motivos acima elencados, ainda há uma dificuldade para identificar as razões determinantes que levaram a uma performance melhor da RNC_1 em comparação às outras. Entretanto, pode-se concluir, de forma geral, que todos os modelos apresentam bons resultados, podendo ser efetivamente utilizados para a realização de previsões em outras imagens.

O código em linguagem Python utilizado nesta etapa do trabalho pode ser encontrado no Apêndice VI deste trabalho.

4.4 Parte 4 - Interpretação dos resultados das redes neurais convolucionais

Um dos grandes desafios da área de aprendizado de máquina é o fato de que, apesar dos modelos gerados serem capazes de fazer previsões precisas em problemas de classificação ou de regressão, muitas vezes não é compreensível ou facilmente explicável para uma pessoa os resultados produzidos pelo modelo, ou seja, como o algoritmo conseguiu chegar a um determinado resultado partindo dos dados iniciais. Isso decorre do fato de muitos algoritmos de aprendizado de máquina serem denominados algoritmos do tipo “caixa preta” ou *blackbox*, ou seja, são sistemas altamente complexos, em que não se entende seu funcionamento interno e a razão de gerarem determinado resultado (Meske *et al.*, 2022).

Nesse contexto, existe uma área da inteligência artificial que busca justamente interpretar e explicar de melhor forma para humanos os resultados de um modelo, sendo que essa área é denominada de Inteligência Artificial Explicável ou, em inglês, *Explainable Artificial Intelligence* (xAI) (Laato *et al.*, 2022).

Segundo Haque *et al.* (2023), as redes neurais são um tipo desse sistema complexo e difícil de ser interpretado. Haar *et al.* (2023), por sua vez, também afirmaram que as

redes neurais convolucionais são modelos de difícil entendimento, existindo diversos métodos para explicar os resultados desse algoritmo de visão computacional. No caso das redes neurais, tem-se que apesar do algoritmo e do seu processo de treinamento ser de fácil entendimento, após o modelo ser treinado, a dificuldade de interpretação se deve ao fato de que há uma combinação complexa entre os neurônios de cada camada da rede com os da camada seguinte, de forma que se torna de difícil entendimento qual informação seria determinante para o resultado final da rede ou como a variação dos dados mudaria a resposta do algoritmo. Nas redes neurais convolucionais, também há problemas de se interpretar as operações que os filtros treinados realizam na imagem original.

Qin *et al.* (2018) afirmaram que uma das formas de interpretar os resultados de uma rede neural convolucional seria através de *feature maps* ou mapas de características. Os *feature maps* são etapas intermediárias do processo de classificação ou de regressão de uma rede neural convolucional já treinada, após a imagem original ser submetida a uma camada de convolução (filtros que destacam as principais características da imagem) ou das camadas de *pooling* (que reduzem o tamanho original das imagens, originais ou já filtradas). A análise desses mapas de características permite que se tenha um entendimento básico de como o modelo está tomando suas decisões, bem como quais características ou elementos da imagem original são os mais importantes para o resultado final fornecido.

Dessa forma, como forma de compreender melhor o processo de tomada de decisão das redes neurais convolucionais treinadas com imagens provenientes de uma simulação BIM 4D, será feita uma avaliação dos *feature maps* gerados pelo modelo. Para tanto, serão selecionadas 3 imagens aleatoriamente do banco de dados de imagens desenvolvido nessa pesquisa e as mesmas serão submetidas à RNC_1, que foi a rede que obteve os melhores resultados estatísticos. Serão gerados os mapas de características da imagem após ela ser submetida à primeira camada de convolução (6 filtros), à primeira camada de *pooling* e, por fim, à segunda camada de convolução (16 filtros). O código em Python para a geração dos *feature maps* pode ser acessado no site < <https://towardsdatascience.com/convolutional-neural-network-feature-map-and-filter-visualization-f75012a5a49c> > (acesso dezembro de 2022).

A Figura 26 trata da primeira imagem a ser analisada, sendo que os *feature maps* gerados estão nas Figuras 27, 28 e 29.

Figura 26: Primeira imagem analisada



Figura 27: *Feature maps* da primeira imagem analisada após a primeira camada de convolução

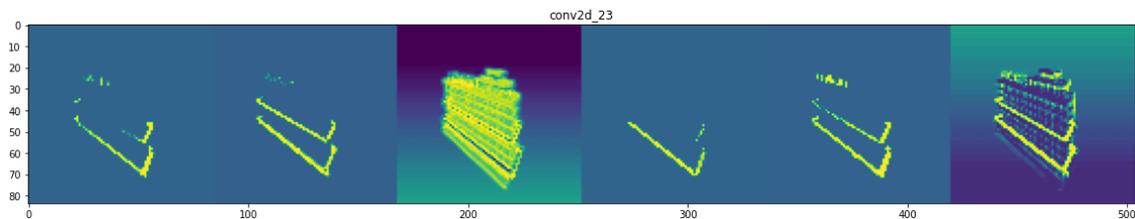


Figura 28: *Feature maps* da primeira imagem analisada após a primeira camada de *pooling*

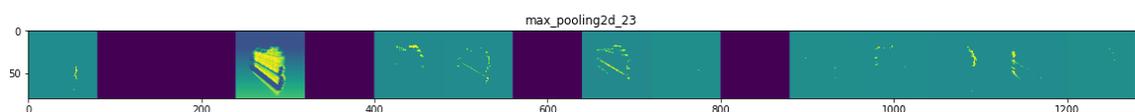
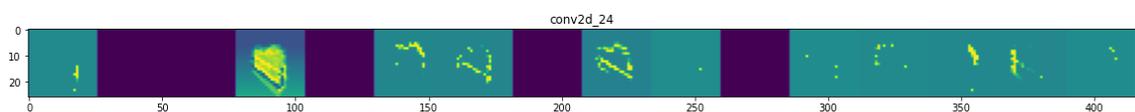


Figura 29: *Feature maps* da primeira imagem analisada após a segunda camada de convolução



Como se pode notar, na imagem da Figura 26, mostra-se uma imagem da simulação da construção em uma etapa final da obra, em que apenas parte da estrutura da cobertura necessita ser construída, sendo que estão presentes na imagem diversos elementos estruturais, bem como as bandejas do segundo e do quarto pavimentos e as fôrmas dos pilares da cobertura.

Ao ser submetida à primeira camada de convolução, que aplica filtros à imagem inicial, gerando 6 imagens distintas (Figura 27), nota-se em um primeiro momento que a rede neural convolucional deu um maior destaque tanto para as bandejas como para as fôrmas dos pilares, que aparecem de forma destacada em 4 das 6 imagens geradas. Os elementos estruturais, por sua vez, somente são visíveis em 2 dos 6 mapas de características gerados, o que mostra que a RNC_1, no processo de avaliação das imagens para regressão, aparentemente confere um peso maior da sua decisão aos elementos temporários do canteiro, como fôrmas e bandejas, do que elementos estruturais permanentes.

Ao se fazer uma análise das Figuras 28 e 29, percebe-se que já se torna complexa a avaliação das imagens geradas, uma vez que há uma redução da dimensão das imagens, o que dificulta a identificação dos elementos da simulação. Após a segunda camada de convolução (Figura 29), em que há a aplicação de filtros sobre as imagens de dimensão reduzida, se torna ainda mais complexo a identificação de quais elementos estão sendo destacados pela rede.

O próximo passo é analisar a segunda imagem selecionada aleatoriamente para se fazer a avaliação dos *feature maps*, que estão dispostos nas Figura 30, 31, 32 e 33. No caso da imagem selecionada dessa vez, nota-se uma fase intermediária da construção do edifício. Mais uma vez, nota-se a presença da bandeja do primeiro piso, bem como algumas fôrmas de pilares em pavimentos superiores.

Figura 30: Segunda imagem analisada

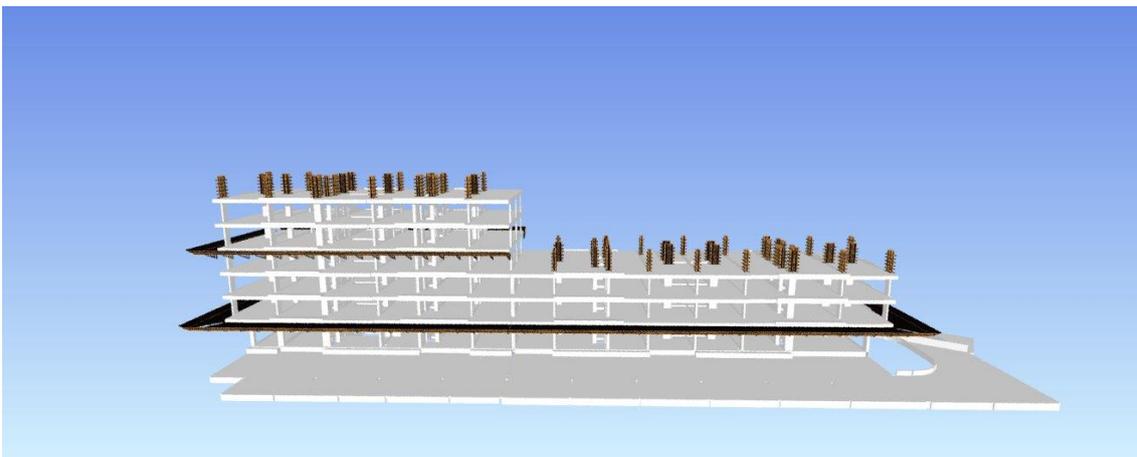


Figura 31: *Feature maps* da segunda imagem analisada após a primeira camada de convolução

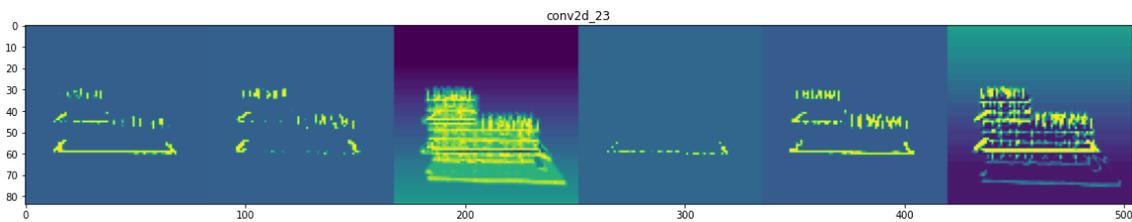


Figura 32: *Feature maps* da segunda imagem analisada após a primeira camada de *pooling*

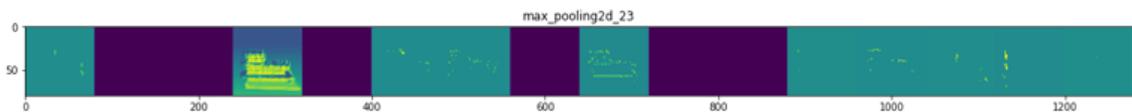
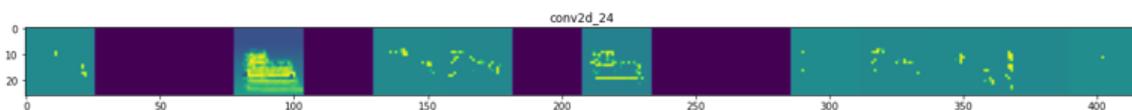


Figura 33: *Feature maps* da segunda imagem analisada após a segunda camada de convolução



Da análise dos mapas de características gerados, pode-se perceber novamente que a rede deu um maior destaque para os elementos temporários da construção, como bandejas e fôrmas, do que para elementos estruturais como pilares, vigas e lajes. Além disso, percebe-se novamente a dificuldade para se interpretar as imagens geradas após a primeira camada de convolução, devido à menor dimensão das imagens e a aplicação de diversos filtros seguidos.

Por fim, analisa-se mais uma imagem de uma etapa intermediária da construção (Figura 34), com o conjunto de *feature maps* gerados (Figuras 35, 36 e 37).

Figura 34: Terceira imagem analisada

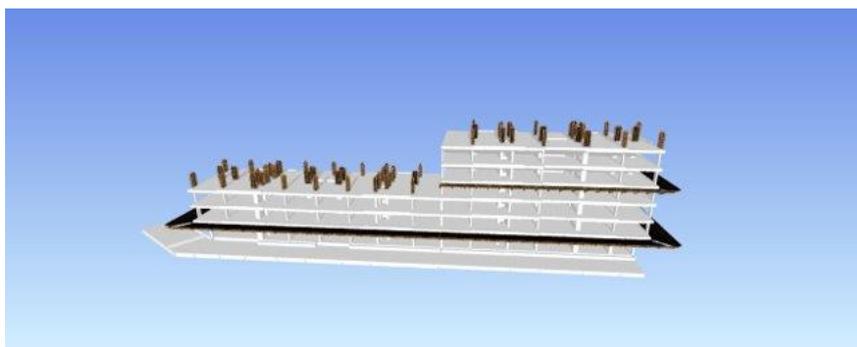


Figura 35: Feature maps da terceira imagem analisada após a primeira camada de convolução

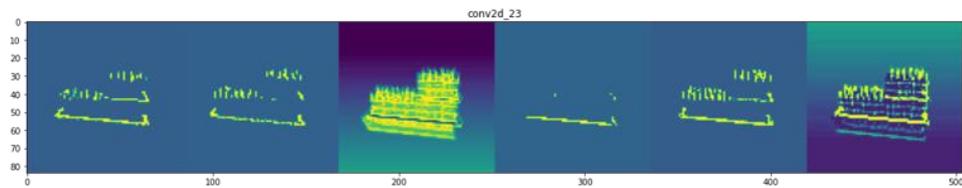


Figura 36: Feature maps da terceira imagem analisada após a primeira camada de pooling

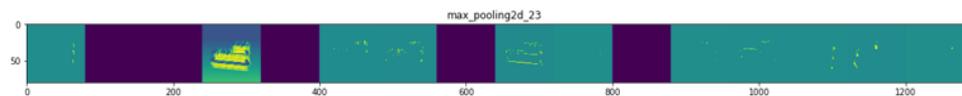
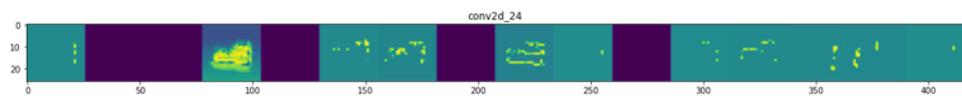


Figura 37: Feature maps da terceira imagem analisada após a segunda camada de convolução



Como se pode notar mais uma vez, os *feature maps* mostram de forma destacada os elementos temporários do canteiro após a primeira camada de convolução (Figura 35), enquanto os elementos estruturais não ficam tão em evidência. Novamente, tem-se que a partir da camada de *max pooling*, torna-se demasiado complexo avaliar o que a rede neural convolucional está processando, uma vez que os elementos restantes das figuras não ficam tão evidentes.

Dessa forma, pode-se presumir que, no caso da RNC_1, que foi o modelo que obteve os melhores resultados de MAE e RMSE para a avaliação das imagens da simulação, o algoritmo busca fazer a regressão para cálculo do progresso da construção utilizando como elementos principais não os elementos estruturais do edifício, como pilares, vigas e lajes, mas sim os elementos temporários do canteiro, como bandejas e fôrmas, e a posição relativa desses elementos na imagem.

Entretanto, tem-se que apesar do algoritmo aparentemente dar um maior destaque a esses elementos temporários para fazer a regressão, também se utiliza dos elementos estruturais nas imagens, apesar de ser em uma menor proporção, uma vez que os pilares, vigas e lajes aparecem em 2 dos 6 filtros gerados após a primeira camada de convolução.

5 CONCLUSÕES

Essa pesquisa teve como objetivo avaliar a integração entre o BIM e a inteligência artificial, com aplicação na área do controle do progresso de construções. Neste estudo, foram treinadas redes neurais convolucionais com imagens de simulações BIM 4D e analisou-se a precisão dos modelos gerados por meio de técnicas estatísticas, além de se realizar uma análise mais aprofundada dos resultados por meio de *feature maps*.

Essa pesquisa apresentou algumas propostas distintas dos estudos normalmente realizados sobre a integração BIM-IA na área do monitoramento de construções. Primeiramente, as redes neurais convolucionais treinadas foram utilizadas para solucionar problemas de regressão, e não de classificação, uma abordagem pouco explorada na literatura científica. Um segundo aspecto singular desta pesquisa foi o treinamento dos modelos de aprendizado de máquina com imagens e informações provenientes da plataforma BIM, sendo que, normalmente, os dados são capturados do ambiente externo e então transferidos à plataforma BIM. Dessa forma, tem-se que esse estudo abordou um fluxo de informações em um sentido contrário ao comumente explorado nas pesquisas científicas.

Quanto ao banco de dados desenvolvido para a execução dessa pesquisa, pôde-se verificar a facilidade que o mesmo foi criado com a exploração de um software BIM 4D, que integrou o modelo tridimensional da construção com o respectivo cronograma, o que fez com que 1200 imagens pudessem ser geradas para o treinamento das redes neurais convolucionais. O progresso da construção em cada imagem, entretanto, foi calculado de forma manual com base em planilhas orçamentárias, de forma que inviabilizaria a criação de um banco de dados muito maior, sendo esse um aspecto que pode ser automatizado em pesquisas futuras.

Esse estudo também mostrou a viabilidade de se utilizar redes neurais convolucionais para a solução de problemas de regressão, além de mostrar que as mesmas podem obter resultados precisos quando treinadas com imagens provenientes de um modelo BIM. As RNCs desenvolvidas nessa pesquisa apresentaram um Erro Médio Absoluto mínimo de 1,93% e máximo de 3,10% em dados de teste, e um mínimo de 0,82% e máximo de 2,28% para os dados de treinamento. Esses resultados mostram não somente que o algoritmo de aprendizado de máquina conseguiu captar as principais características das imagens e relacionar esses atributos com o efetivo progresso da construção, como também mostraram que não houve problemas de *underfitting* e

tampouco de *overfitting*, ou seja, não houve um ajuste excessivo aos dados de treinamento, de forma que os dados de teste pudessem ser avaliados sem enviesamento.

Outra conclusão interessante com relação às redes neurais convolucionais treinadas se deve ao fato de que a RNC com um maior número de parâmetros treináveis foi a que se mostrou mais imprecisa, enquanto a rede que apresentou um menor número de parâmetros treináveis se mostrou a mais precisa. Dessa forma, pode-se afirmar que, no caso dessa pesquisa, não houve uma relação direta entre a complexidade de uma RNC para regressão e os resultados por ela obtidos, ou seja, não necessariamente uma RNC mais complexa, mais robusta e maior fornecerá os melhores resultados. Isso pode ser explicado pelo fato de que uma rede mais simples pode ser capaz de generalizar mais os dados, conseguindo extrair características gerais das imagens fornecidas para treinamento, enquanto uma rede mais complexa pode captar muitos detalhes das imagens treinadas, não conseguindo generalizar bem para outras imagens, o que geraria imprecisões nos resultados. Nesse contexto, uma possibilidade para aumentar a precisão do modelo em pesquisas futuras seria o treinamento de redes neurais convolucionais mais simples, com uma quantidade menor de parâmetros treináveis, através de uma redução do número de camadas ou de neurônios por camada.

Por fim, os *features maps* gerados pelas RNCs permitiu que se pudesse avaliar como o algoritmo de visão computacional analisava as imagens para chegar aos resultados das regressões. Em que pese a técnica utilizada não permita que uma explicação definitiva sobre o funcionamento da rede fosse formada, algumas percepções gerais sobre o funcionamento do algoritmo puderam ser identificadas. De forma geral, foi possível notar que os *feature maps* gerados apresentavam maior destaque para os elementos do canteiro de obras do que para a parte estrutural do edifício, mostrando a importância desses elementos dentro do modelo BIM para a regressão realizada. Dessa forma, tem-se que, apesar de cada banco de dados apresentar suas peculiaridades e cada modelo treinado ser distinto, tem-se que é de fundamental importância que um modelo BIM gerado para controle do progresso de construções deva possuir informações detalhadas sobre o canteiro de obras, inclusive com elementos temporários. Nesse contexto, tem-se que, em pesquisas futuras, o modelo BIM pode conter uma maior quantidade de elementos temporários da construção, como escoramentos e instalações provisórias, o que poderia aumentar a precisão do modelo de aprendizado de máquina treinado.

Dessa forma, foi possível verificar que os objetivos gerais e específicos do trabalho foram alcançados, uma vez que o banco de dados de imagens de uma simulação BIM 4D foi criado; desenvolveram-se 3 estruturas distintas de redes neurais convolucionais para regressão, sendo que as 3 RNCs foram treinadas e avaliadas, apresentando resultados bastante precisos. Por fim, foram analisados os *feature maps* gerados pelas RNCs, de forma que foi possível obter uma melhor explicação sobre os resultados gerados pelo modelo de visão computacional.

Como contribuição científica, tem-se que esse trabalho demonstrou a possibilidade de se integrar redes neurais convolucionais para regressão com imagens geradas por meio de uma simulação BIM 4D. A pesquisa foi aplicada em um contexto de monitoramento do progresso de construções, entretanto, tem-se que os resultados alcançados podem ser aplicados a diversas outras áreas da construção civil.

6 TRABALHOS FUTUROS

Durante o desenvolvimento dessa pesquisa foi possível identificar outros estudos que poderão ser realizados futuramente, seja uma continuação direta do trabalho realizado, seja a utilização do progresso científico alcançado para a exploração de outras temáticas.

Primeiramente, sugere-se uma continuação desta pesquisa com a inserção de mais detalhes no modelo BIM. Neste estudo, somente foi realizada a simulação BIM 4D com o projeto estrutural do edifício, enquanto o canteiro de obras foi modelado apenas com as fôrmas de pilares e as bandejas ao redor da construção. Dessa forma, se propõe uma modelagem mais detalhada, com uma simulação envolvendo não somente o projeto estrutural, mas também os projetos arquitetônicos e de instalações do edifício. Além disso, propõe-se a modelagem de mais elementos do canteiro de obras, como linhas de vida, fôrmas de vigas, escoramentos, instalações provisórias do canteiro, entre outros, destacando-se o fato de que, como a análise dos *feature maps* permitiu identificar que as redes neurais convolucionais utilizavam principalmente as imagens dos elementos do canteiro para calcular o progresso da construção, o treinamento de um novo modelo com imagens de um canteiro mais detalhado pode resultar em uma rede neural mais precisa. O objetivo dessa modelagem com mais detalhes é expandir os resultados dessa pesquisa, de forma que estes estudos possam ser utilizados futuramente para monitorar o progresso de construções de forma automatizada.

Uma vez que a geração das simulações BIM 4D foi feita de forma manual, outro possível trabalho futuro seria o desenvolvimento de uma rotina que permitisse que as simulações da construção fossem geradas de forma automatizada, o que pode ser feito com recursos de programação visual como o Dynamo. Dessa forma, podem ser gerados bancos de dados mais robustos com menos trabalho manual.

Também se deve ressaltar que esse trabalho mostrou ser possível uma integração entre as redes neurais convolucionais e o BIM 4D. Nesse contexto, sugere-se um conjunto de pesquisas que integrem as redes neurais convolucionais com outras dimensões do BIM, como por exemplo o BIM 5D e o BIM 8D. No caso do BIM 5D, poderia ser verificado a possibilidade de prever o custo de uma obra a partir de imagens. Como relação ao BIM 8D, que trata da segurança no trabalho, poderia ser verificado a possibilidade de se modelar um canteiro de obras em um software BIM e classificar

imagens geradas a partir desse modelo como em conformidade ou não com as normas de segurança no trabalho.

Quanto às redes neurais convolucionais, um trabalho futuro poderia ser a reprodução dessa pesquisa utilizando o método de validação cruzada (*cross-validation*) para avaliação do modelo de aprendizado de máquina, além da utilização de outras estruturas de redes neurais convolucionais. Para identificar as melhores estruturas da rede neural e otimizar seus hiperparâmetros, podem ser utilizadas as técnicas de *grid search* e de *random search*, que permitem que sejam avaliados modelos com diferentes números de camadas e neurônios. A utilização da validação cruzada e o uso de outras estruturas de rede poderiam fornecer uma melhor compreensão sobre a precisão das redes neurais convolucionais para regressão.

Outra possibilidade de pesquisa é o treinamento de um novo modelo utilizando-se não somente imagens, mas também características da construção, como por exemplo número de andares, tipo de estrutura, quantidade de apartamentos, dados temporais, entre outros. Esse tipo de análise pode inclusive ser utilizado para o início da criação de um modelo mais geral, que pode ser aplicado para diversas construções, e não somente uma obra específica.

Por fim, uma outra opção de trabalho futuro seria a reprodução dessa pesquisa com outros tipos de construções, como por exemplo pontes, estradas, casas residenciais e edifícios com diferentes estruturas ou quantidades de pavimentos.

REFERÊNCIAS BIBLIOGRÁFICAS

AHILA PRIYADHARSHINI, Ramar *et al.* Maize leaf disease classification using deep convolutional neural networks. *Neural Computing and Applications*, v. 31, p. 8887-8895, 2019.

AKINOSHO, Taofeek D. *et al.* Deep learning in the construction industry: A review of present status and future innovations. *Journal of Building Engineering*, v. 32, p. 101827, 2020.

ÁLVARES, J. S.; COSTA, D. B. Construction Progress Monitoring Using Unmanned Aerial System and 4D BIM. 27th Annual Conference of the International Group for Lean Construction (IGLC), 2019.

ARIF, F.; KHAN, W. A. A Real-Time Productivity Tracking Framework Using Survey-Cloud-BIM Integration. *Arabian Journal for Science and Engineering*, 2020.

ARNAL, I. P. Why don't we start at the beginning? The Basics of a Project: Lean Planning and Pre-Construction, *BIM News Last trends of the AECO sector*, BIM Community, 2018.

AUTODESK INC, Autodesk Navisworks.
<<https://www.autodesk.com.br/products/navisworks/overview?term=1-YEAR&tab=subscription>>, 2023 (acesso em 29 de janeiro de 2023).

BARROS, L. B. Aplicação de Redes Neurais Artificiais no contexto de estimativa de custos na construção de rodovias. *Dissertação (Mestrado) – Programa de Pós-Graduação em Estruturas e Construção Civil – Universidade de Brasília, Brasília, 2019.*

BECERIK-GERBER, B.; CALIS, G.; JAZIZADEH, F.; LI, N. Application Areas and Data Requirements for BIM-Enabled Facilities Management. *Journal of Construction Engineering and Management* Vol. 138, 2011.

BRASIL, 2020. Decreto nº 10.306, de 2 de abril de 2020. Estabelece a utilização do Building Information Modelling na execução direta ou indireta de obras e serviços de engenharia realizada pelos órgãos e pelas entidades da administração pública federal, no âmbito da Estratégia Nacional de Disseminação do Building Information Modelling - Estratégia BIM BR, instituída pelo Decreto nº 9.983, de 22 de agosto de 2019. Brasília, DF. Diário Oficial da União.

BRASIL, 2021. Lei nº 14.133, de 1º de abril de 2021. Lei de Licitações e Contratos Administrativos. Brasília, DF. Diário Oficial da União.

BRAUN, A.; BORRMANN, A. Combining inverse photogrammetry and BIM for automated labeling of construction site images for machine learning. *Automation in Construction* 106, 2019.

CALOBA, G; KLAES, M. Gerenciamento de projetos com pdca: Conceito e técnicas para o planejamento, monitoramento e avaliação de desempenhos para projetos e portfólios. Alta Books; 1ª edição. 2016.

CÁRDENAS-MONTES, Miguel. Sobreajuste–overfitting. Ciemat (Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas), 2006.

CIREGAN, Dan; MEIER, Ueli; SCHMIDHUBER, Jürgen. Multi-column deep neural networks for image classification. In: 2012 IEEE conference on computer vision and pattern recognition. IEEE, 2012. p. 3642-3649.

DA SILVA, Rodrigo; CARVALHO, Michele. Revisão da literatura da integração BIM-IA na construção civil. 4º Congresso Português de Building Information Modelling, 2022.

DEL PICO, W. J. Project Control: Integrating Cost and Schedule in Construction. Hoboken: John Wiley & Sons, Inc., 2013.

DENG, Hui *et al.* Automatic indoor construction process monitoring for tiles based on BIM and computer vision. *Journal of construction engineering and management*, v. 146, n. 1, p. 04019095, 2020.

DENG, L.; YU, D. Deep Learning: Methods and Applications. *Foundations and Trends® in Signal Processing: Vol. 7: No. 3–4*, pp 197-387., 2014.

EASTMAN, P.; TEICHOLZ; SACKS, L. *BIM Handbook, A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors*, 2nd ed., John Wiley & Sons, Inc., New Jersey, 2011.

EKANAYAKE, Biyanka *et al.* Computer vision-based interior construction progress monitoring: A literature review and future research directions. *Automation in construction*, v. 127, p. 103705, 2021.

ERSHADI, Mahmoud *et al.* Implementation of Building Information Modelling in infrastructure construction projects: A study of dimensions and strategies. *International Journal of Information Systems and Project Management*, v. 9, n. 4, p. 43-59, 2021.

EVERITT, Brian S.; SKRONDAL, Anders. *The Cambridge dictionary of statistics*. 2010.

FAN, Q.; WONG, K. D. Building Information Modelling (BIM) for sustainable design. *Facilities*, 31(3) 2013 138–157, 2013.

FANG, T.; LEI, K. WANG, Z.; WU, T.; YANG, B.; ZHANG, B.; ZHANG, Q. Vision Based Framework for Automatic Progress Monitoring of Precast Walls by Using Surveillance Videos during the Construction Phase. **Journal of Computing in Civil Engineering**, 2021.

FARINHA, R.; HUTTUNEN, H.; LAASONEN, M.; LOMIO, F. Classification of Building Information Model (BIM) Structures with Deep Learning. 2018 7th European Workshop on Visual Information Processing (EUVIP) 26-28 November, 2018, Tampere, Finland, 2018.

FEI, Y.; LI, J. Q.; LIU, Y.; MAHBOUB, K. C.; WANG, K. C. P.; YANG, G.; ZHANG, A. A. Automatic Pavement Type Recognition for Image-Based Pavement Condition Survey Using Convolutional Neural Network. *Journal of Computing in Civil Engineering*, 2021.

FULFORD, Richard; STANDING, Craig. Construction industry productivity and the potential for collaborative practice. *International journal of project management*, v. 32, n. 2, p. 315-326, 2014.

GOLPARVAR-FARD, Mani; SAVARESE, Silvio; PEÑA-MORA, Feniosky. Automated model-based recognition of progress using daily construction photographs and IFC-based 4D models. In: *Construction Research Congress 2010: Innovation for Reshaping Construction Practice*. 2010. p. 51-60.

GÖYMEN, S.; GÜÇLÜER, K.; GÜNAYDIN, O.; ÖZBEYAZ, A. A comparative investigation using machine learning methods for concrete compressive strength estimation. *Materials Today Communications* 27, 2021.

HAAR, Lynn Vonder; ELVIRA, Timothy; OCHOA, Omar. An analysis of explainability methods for convolutional neural networks. *Engineering Applications of Artificial Intelligence*, v. 117, p. 105606, 2023.

HAQUE, AKM Bahalul; ISLAM, AKM Najmul; MIKALEF, Patrick. Explainable Artificial Intelligence (XAI) from a user perspective: A synthesis of prior literature and problematizing avenues for future research. *Technological Forecasting and Social Change*, v. 186, p. 122120, 2023.

HAUTALA, Kirsi; JÄRVENPÄÄ, Matti-Esko; PULKKINEN, Pekka. Digitalization transforms the construction sector throughout asset's life-cycle from design to operation and maintenance. *Stahlbau*, v. 86, n. 4, p. 340-345, 2017.

HAYKIN, S. *Redes Neurais: Princípios e Prática*. Porto Alegre RS: Bookman, 2001.

HE, W.; LI, P.; LI, Y.; SHEN, J.; XIONG, X.; ZHENG, X. Detecting safety helmet wearing on construction sites with bounding-box regression and deep transfer learning. *Computer-Aided Civil and Infrastructure Engineering*, 2021.

HEATON, Jeff. Ian Goodfellow, Yoshua Bengio, and Aaron Courville: *Deep learning: The MIT Press*, 2016, 800 pp, ISBN: 0262035618. *Genetic Programming and Evolvable Machines*, v. 19, n. 1-2, p. 305-307, 2018.

HINTON, Geoffrey; SEJNOWSKI, Terrence J. (Ed.). *Unsupervised learning: foundations of neural computation*. MIT press, 1999.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. ISO 19650-1 Organization and digitization of information about buildings and civil engineering works, including building information modelling (BIM) — Information management using building information modelling — Part 1: Concepts and principles, 2018.

ISO, BSEN. 19650-1; Organization and Digitization of Information about Buildings and Civil Engineering Works, Including Building Information Modelling (BIM). *Information Management Using Building Information Modelling-Concepts and Principles*. British Standard: London, UK, 2018.

JAISWAL, Sunil *et al.* *Enabling Quality in Lean Construction: Integrating the Principles of Total Quality Management with 9D-BIM*. 2022.

JUNG, R.; KOO, B.; YU, Y. Automatic classification of wall and door BIM element subtypes using 3D geometric deep neural networks. *Advanced Engineering Informatics* 47, 2021.

KAELBLING, Leslie Pack; LITTMAN, Michael L.; MOORE, Andrew W. Reinforcement learning: A survey. *Journal of artificial intelligence research*, v. 4, p. 237-285, 1996.

KAMARDEEN, I. 8D BIM modelling tool for accident prevention through design. Egbu, C. (Ed) Procs 26th Annual ARCOM Conference, 6-8 September 2010, Leeds, UK, Association of Researchers in Construction Management, 281-289, 2010.

KARUNASINGHA, Dulakshi Santhusitha Kumari. Root mean square error or mean absolute error? Use their ratio as well. Information Sciences, v. 585, p. 609-629, 2022.

KHALLAF, Rana; KHALLAF, Mohamed. Classification and analysis of deep learning applications in construction: A systematic literature review. Automation in construction, v. 129, p. 103760, 2021.

KOHAVI, R.; PROVOST, F. Glossary of Therms, 1998.

KOUIDER, Tahar; SYKES, Paul; HAMMA-ADAMA, Mansur. BIM manager, coordinator, consultant, analyst..., what does a confused AEC industry need?. 2019.

KOUTAMANIS, Alexander. Dimensionality in BIM: Why BIM cannot have more than four dimensions?. Automation in Construction, v. 114, p. 103153, 2020.

KRIZHEVSKY, Alex; SUTSKEVER, Ilya; HINTON, Geoffrey E. Imagenet classification with deep convolutional neural networks. Communications of the ACM, v. 60, n. 6, p. 84-90, 2017.

KROPP, Christopher; KOCH, Christian; KÖNIG, Markus. Model-based pose estimation for visual indoor progress monitoring using line features. In: Proceedings of the 16th International Conference on Computing in Civil and Building Engineering. 2016.

LAATO, Samuli *et al.* How to explain AI systems to end users: a systematic literature review and research agenda. Internet Research, v. 32, n. 7, p. 1-31, 2022.

LAWRENCE, Steve *et al.* Face recognition: A convolutional neural-network approach. IEEE transactions on neural networks, v. 8, n. 1, p. 98-113, 1997.

LECUN, Yann *et al.* Gradient-based learning applied to document recognition. Proceedings of the IEEE, v. 86, n. 11, p. 2278-2324, 1998.

LIN, Z. Y.; MA, Z. L.; PETZOLD, F. A Real-Time 4D Augmented Reality System for Modular Construction Progress Monitoring. 36th International Symposium on Automation and Robotics in Construction, 2019.

MAALEK, Reza; RUWANPURA, Janaka; RANAWEERA, Kamal. Evaluation of the state-of-the-art automated construction progress monitoring and control systems. In: Construction Research Congress 2014: Construction in a Global Network. 2014. p. 1023-1032.

MAINI, DEEPAK. Up and Running with Autodesk Navisworks 2023 (English Edition). 2023.

MATOS, C. R. O Uso do BIM na Fiscalização de Obras Públicas. Dissertação (Mestrado) – Programa de Pós-Graduação em Estruturas e Construção Civil – Universidade de Brasília, Brasília, 2016.

MATPLOTLIB ORG. < <https://matplotlib.org/> >, 2023. Acesso em 29 de janeiro de 2023.

MATSUGU, Masakazu *et al.* Subject independent facial expression recognition with robust face detection using a convolutional neural network. Neural Networks, v. 16, n. 5-6, p. 555-559, 2003.

MATTOS, Aldo Dórea. Planejamento e controle de obras. Oficina de Textos, 2019.

MCCULLOCH, Warren S.; PITTS, Walter. A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics, v. 5, p. 115-133, 1943.

MESKE, Christian *et al.* Explainable artificial intelligence: objectives, stakeholders, and future research opportunities. Information Systems Management, v. 39, n. 1, p. 53-63, 2022.

MOSTAFA, K.; HEGAZY, T. Review of image-based analysis and applications in construction. *Automation in Construction*, v. 122, 2021.

NAVON, R.; SACKS, R.: Assessing research issues in automated Project performance control (APPC). *Automation in Construction*, 16(4), 474–484, 2007.

NumPy, Org. < <https://numpy.org/> >, 2023. Acesso em 29 de janeiro de 2023.

OPENCV ORG. < <https://opencv.org/about/> >, 2023. Acesso em 29 de janeiro de 2023.

PAN, Yue; ZHANG, Limao. A BIM-data mining integrated digital twin framework for advanced project management. *Automation in Construction*, v. 124, p. 103564, 2021.

PANDAS. < <https://pandas.pydata.org/about/> >, 2023. Acesso em 29 de janeiro de 2023.

PANERU, Suman; JEELANI, Idris. Computer vision applications in construction: Current state, opportunities & challenges. *Automation in Construction*, v. 132, p. 103940, 2021.

PENG, Changhai. Calculation of a building's life cycle carbon emissions based on Ecotect and building information modeling. *Journal of Cleaner Production*, v. 112, p. 453-465, 2016.

PETERS, E. BIM and Geospatial Information System. In UNDERWOOD, J.; ISIKDAG, U. *Handbook of Research on Building Information Modeling and Construction Informatics: Concepts and Technologies*. New York: Information Science Reference, 2009.

PYTHON ORG. < <https://www.python.org/about/apps/> >, 2023. Acesso em 29 de janeiro de 2023.

QIN, Zhuwei *et al.* How convolutional neural network see the world-A survey of convolutional neural network visualization methods. arXiv preprint arXiv:1804.11191, 2018.

QURESHI, Abdul Hannan *et al.* Factors impacting the implementation process of automated construction progress monitoring. *Ain Shams Engineering Journal*, v. 13, n. 6, p. 101808, 2022.

RAHIMIAN, Farzad Pour *et al.* On-demand monitoring of construction projects through a game-like hybrid application of BIM and machine learning. *Automation in Construction*, v. 110, p. 103012, 2020.

REED, Russell; MARKSII, Robert J. *Neural smithing: supervised learning in feedforward artificial neural networks.* Mit Press, 1999.

REJA, Varun Kumar; VARGHESE, Koshy; HA, Quang Phuc. Computer vision-based construction progress monitoring. *Automation in Construction*, v. 138, p. 104245, 2022.

RUSSEL, S.; NORVIG, P. *Artificial Intelligence: A Modern Approach (4th Edition)*, 2020.

SCIKIT-LEARN org. < <https://scikit-learn.org/stable/#> >, 2023. Acesso em 29 de janeiro de 2023.

SHANTI, Mohammad Z. *et al.* Real-time monitoring of work-at-height safety hazards in construction sites using drones and deep learning. *Journal of safety research*, v. 83, p. 364-370, 2022.

SIMON, P. *Too Big to Ignore: The Business Case for Big Data.* Wiley, 2015.

SMITH, Peter. BIM implementation—global strategies. *Procedia engineering*, v. 85, p. 482-492, 2014.

TANG, Jinjun *et al.* An improved fuzzy neural network for traffic speed prediction considering periodic characteristic. *IEEE Transactions on Intelligent Transportation Systems*, v. 18, n. 9, p. 2340-2350, 2017.

TENSORFLOW ORG. < <https://www.tensorflow.org/?hl=pt-br> >, 2023. Acesso em 29 de janeiro de 2023.

UTKUCU, Duygu; SÖZER, Hatice. Interoperability and data exchange within BIM platform to evaluate building energy performance and indoor comfort. *Automation in Construction*, v. 116, p. 103225, 2020.

UZAIR, Muhammad; JAMIL, Noreen. Effects of hidden layers on the efficiency of neural networks. In: 2020 IEEE 23rd international multitopic conference (INMIC). IEEE, 2020. p. 1-6.

VALUEVA, M. V.; NAGORNOV, N. N.; LYAKHOV, P. A.; VALUEV, G. V.; CHERVYAKOV, N. I. Application of the residue number system to reduce hardware costs of the convolutional neural network implementation. *Mathematics and Computers in Simulation* Volume 177, 2020.

XU, Yuesheng; ZHANG, Haizhang. Convergence of deep convolutional neural networks. *Neural Networks*, v. 153, p. 553-563, 2022.

WEI, Wei *et al.* Integrated vision-based automated progress monitoring of indoor construction using mask region-based convolutional neural networks and BIM. *Automation in Construction*, v. 140, p. 104327, 2022.

ZABIN, Asem *et al.* Applications of machine learning to BIM: A systematic literature review. *Advanced Engineering Informatics*, v. 51, p. 101474, 2022.

ZHOU, Y.-T. *et al.* Image restoration using a neural network. *IEEE transactions on acoustics, speech, and signal processing*, v. 36, n. 7, p. 1141-1151, 1988.

APÊNDICE I

Figura 38: Simulação 1



Figura 39: Simulação 2

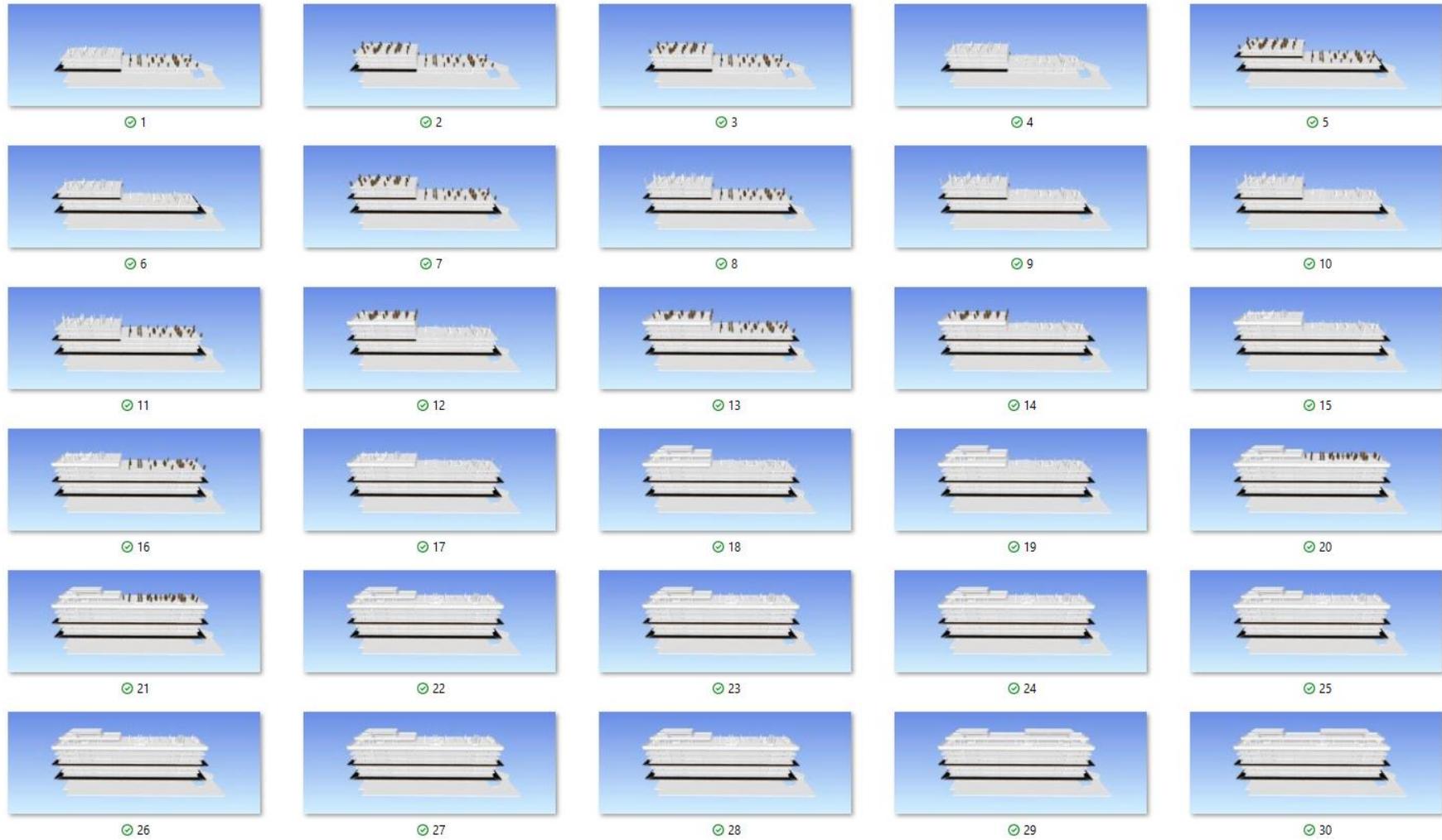


Figura 40: Simulação 3

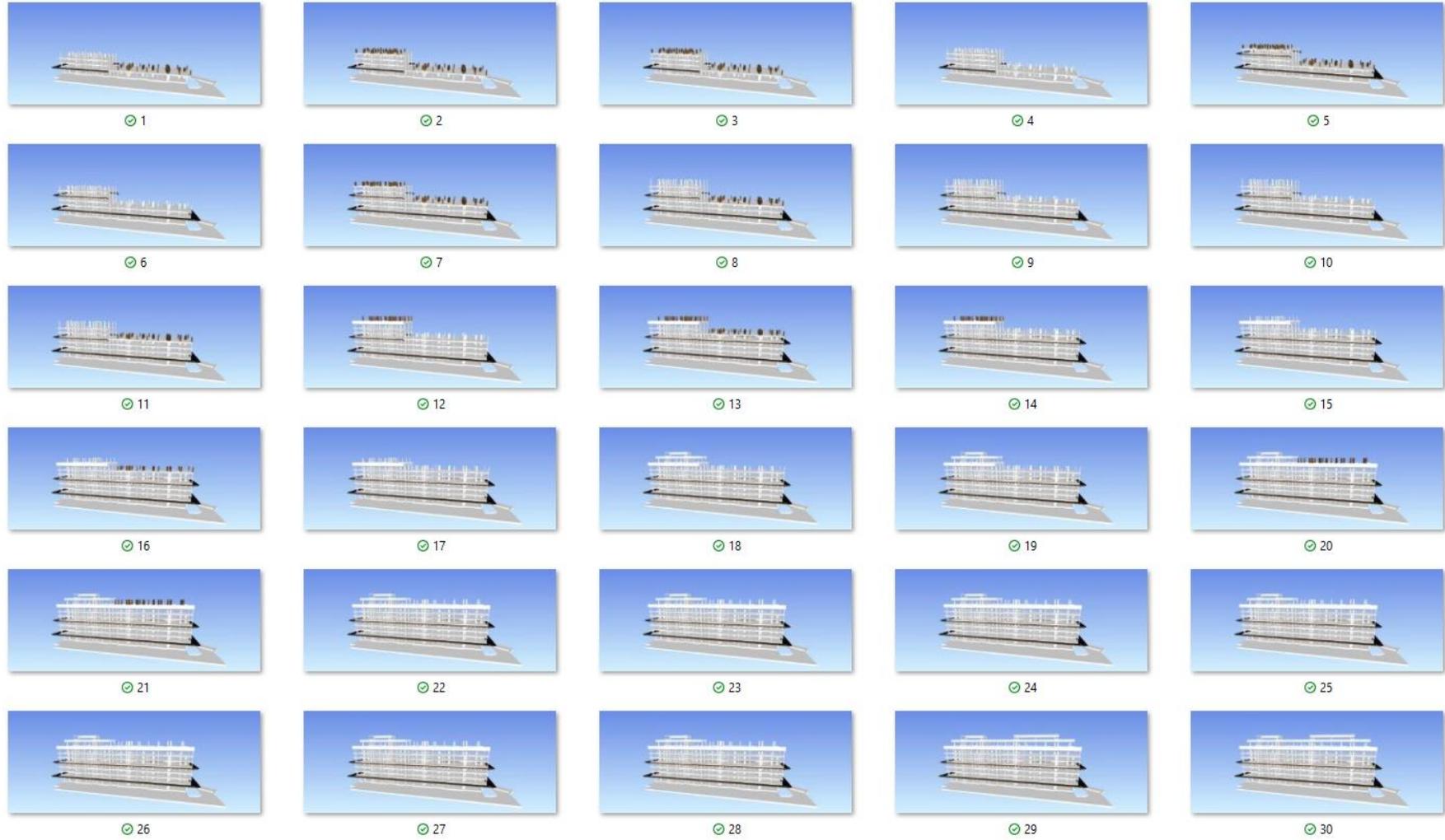


Figura 41: Simulação 4

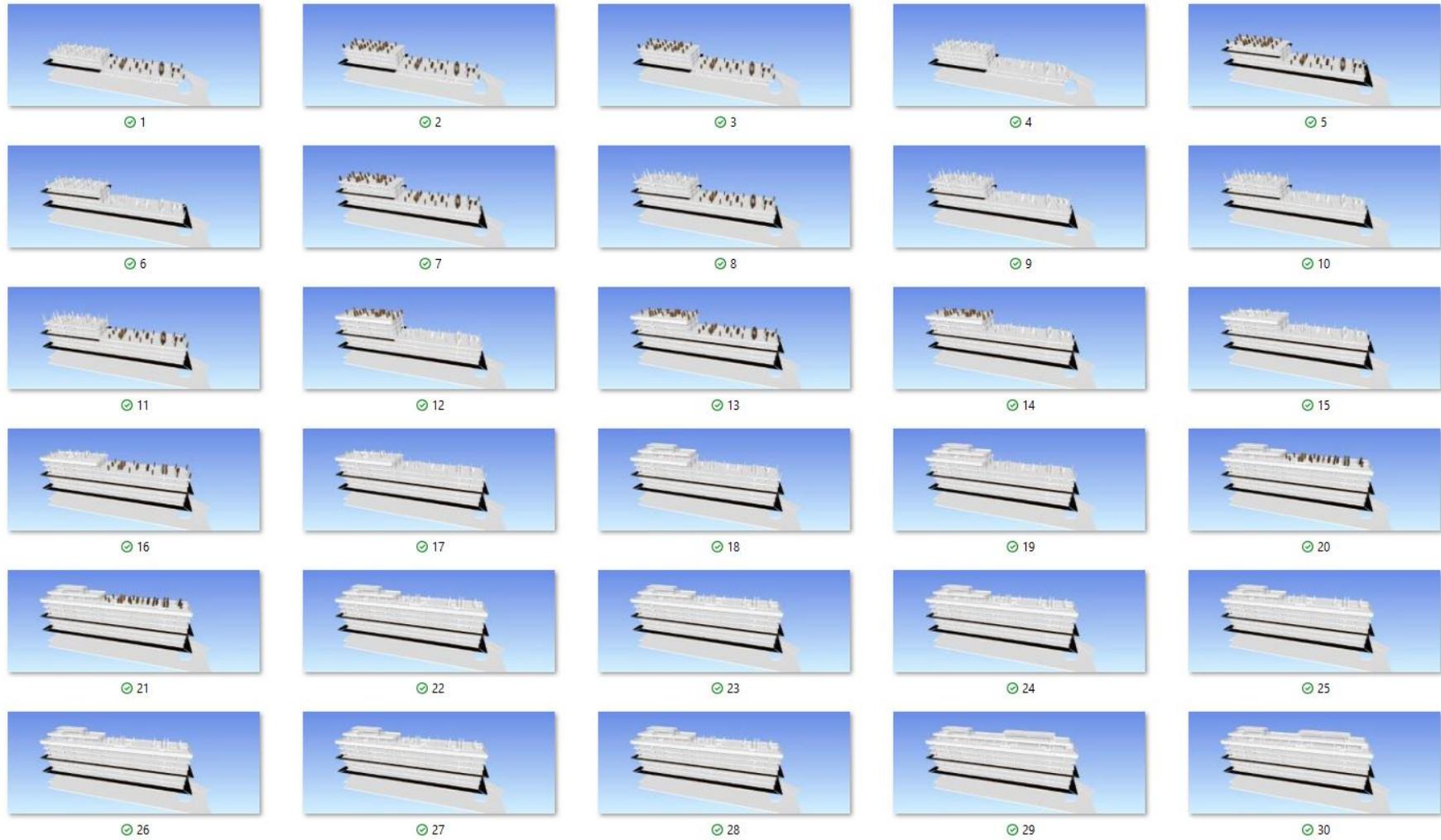


Figura 42: Simulação 5



Figura 43: Simulação 6

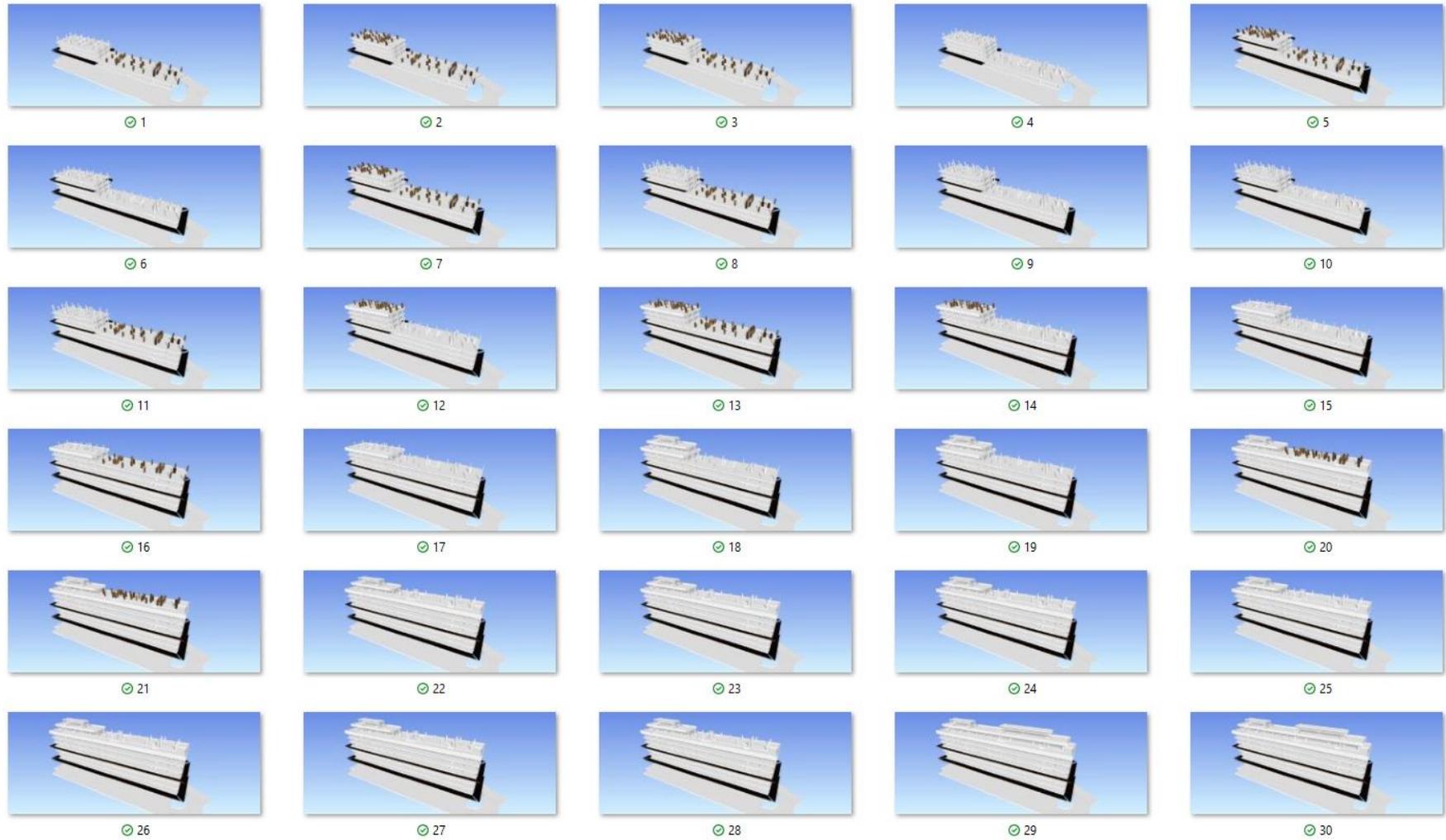


Figura 44: Simulação 7



Figura 45: Simulação 8

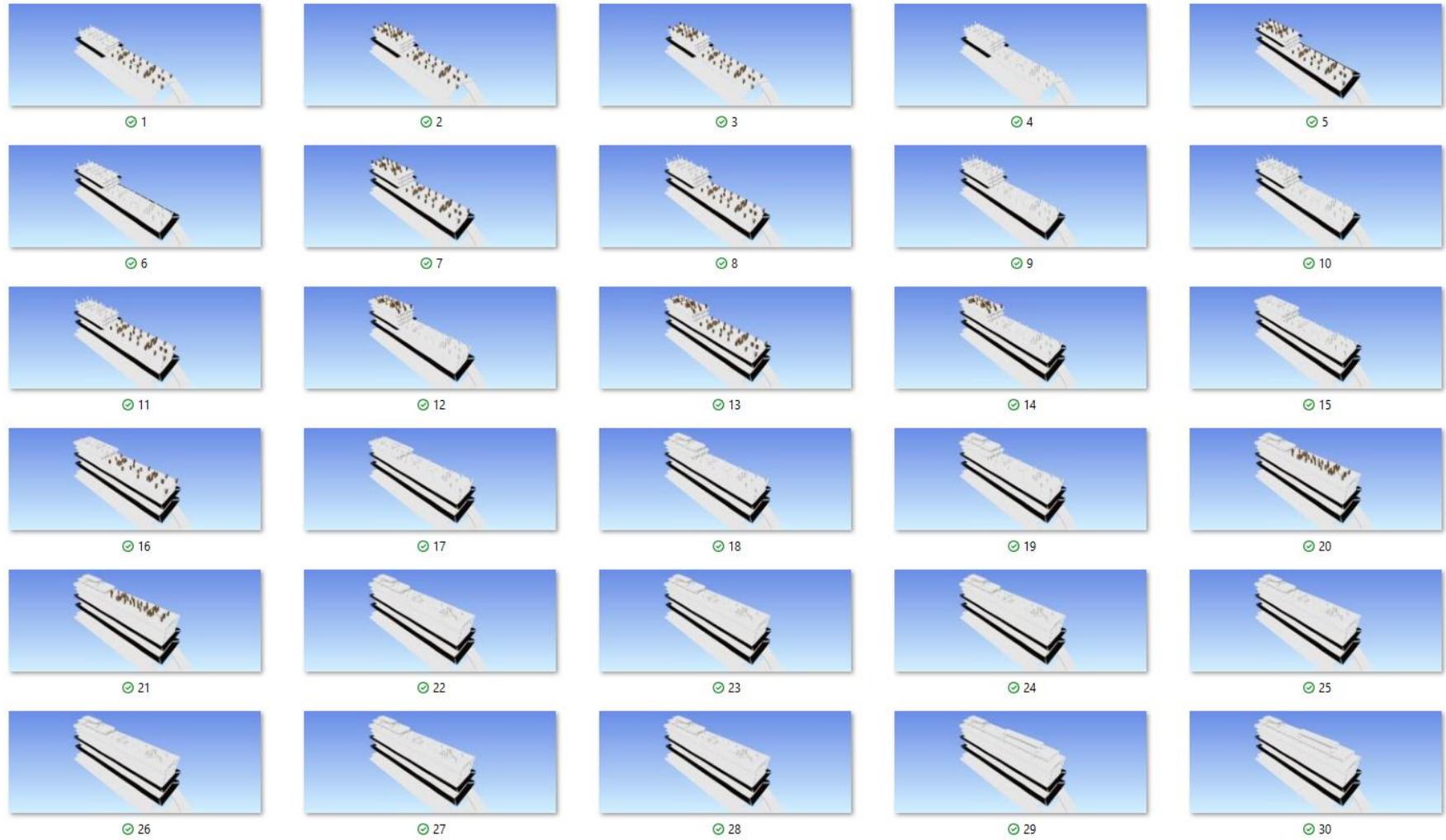


Figura 46: Simulação 9



Figura 47: Simulação 10

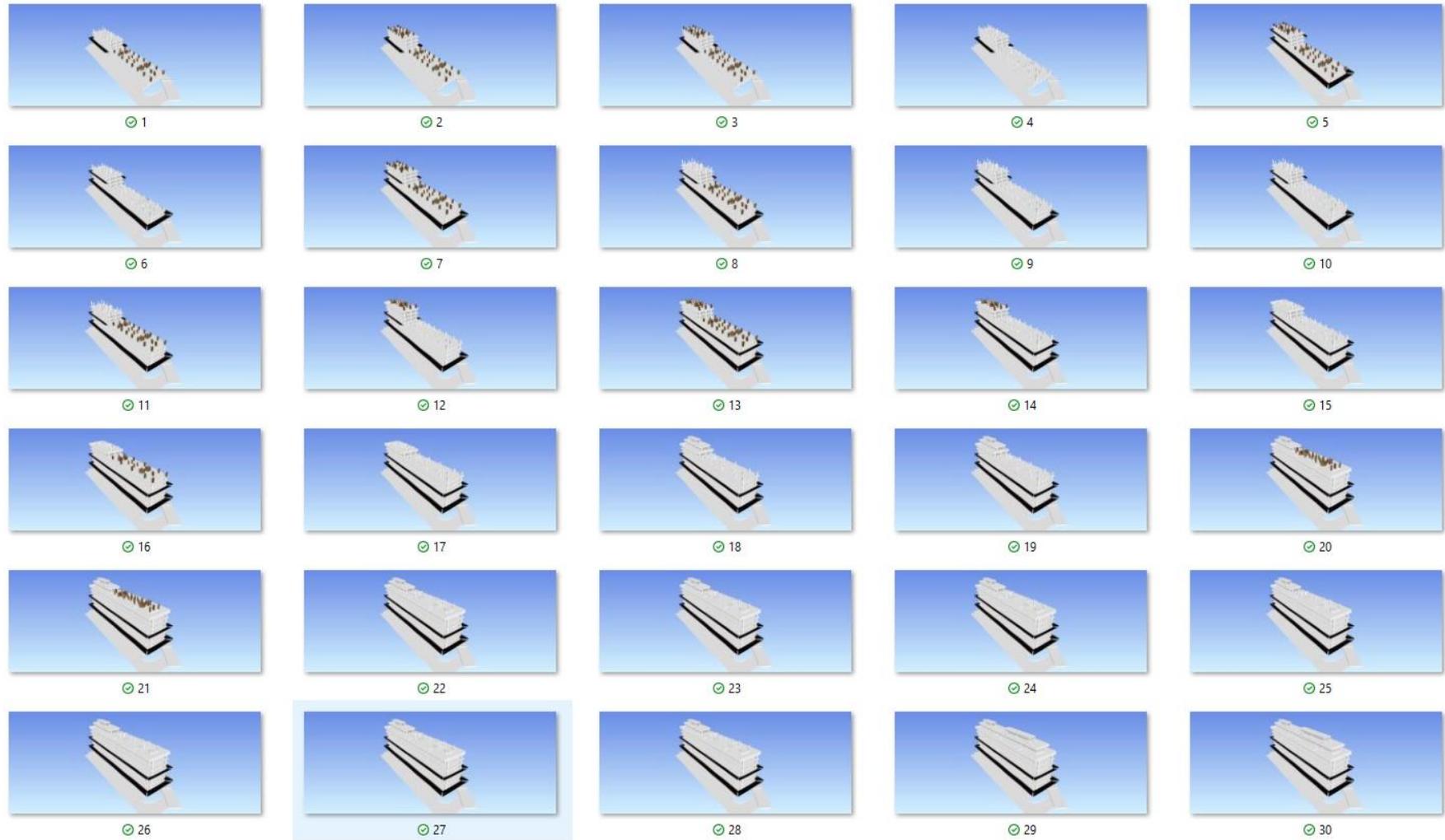


Figura 48: Simulação 11

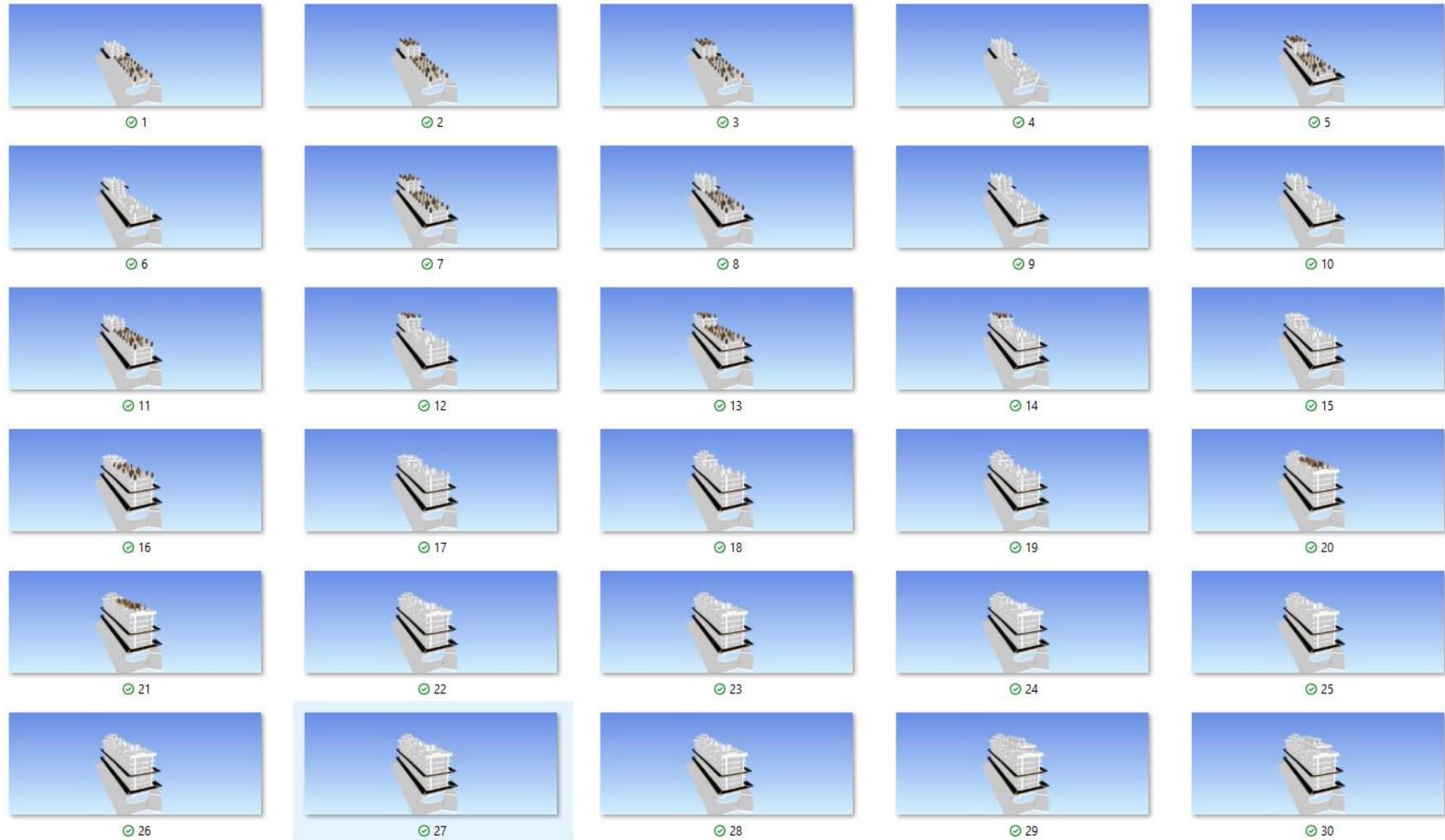


Figura 49: Simulação 12



Figura 50: Simulação 13

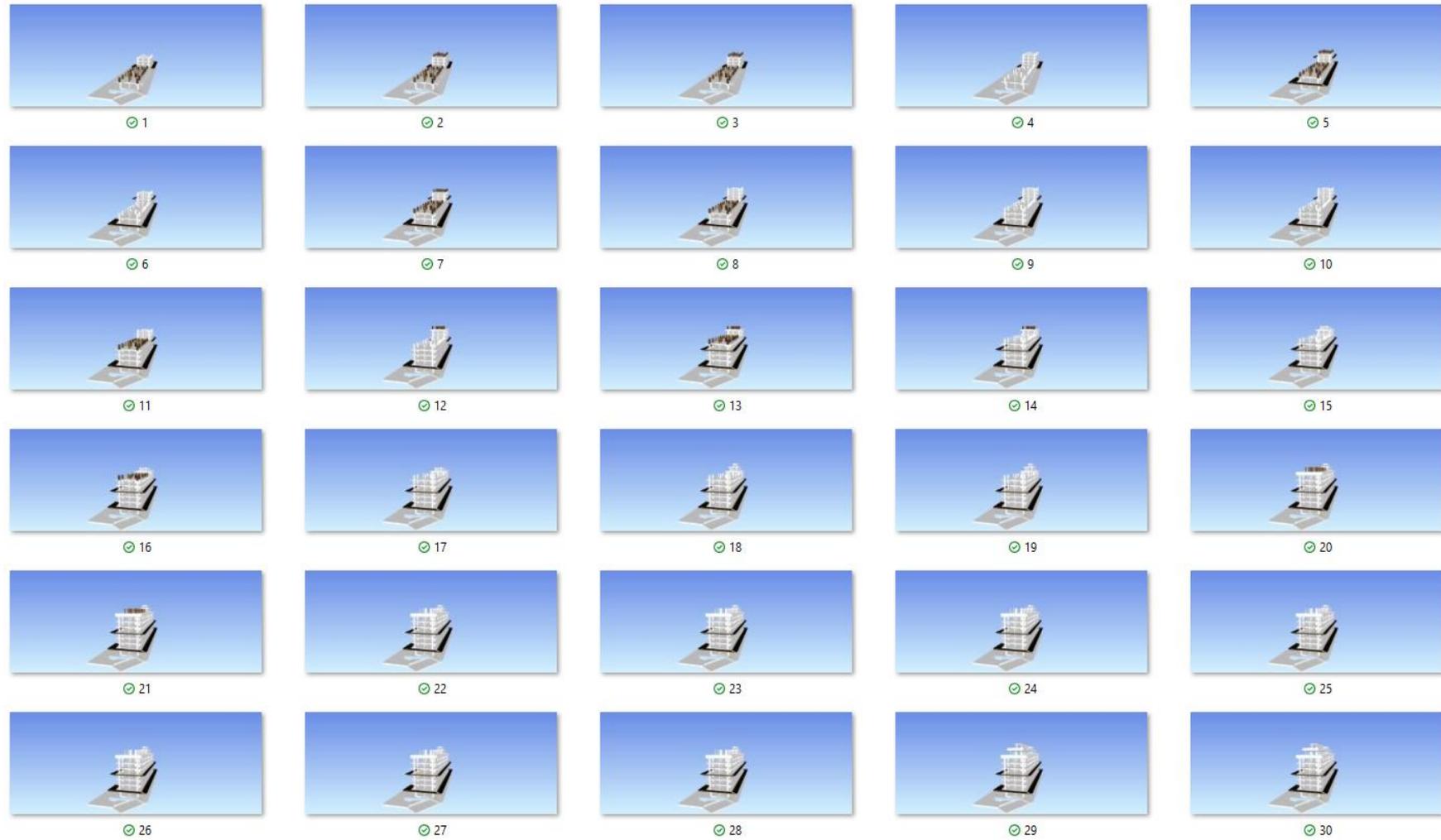


Figura 51: Simulação 14

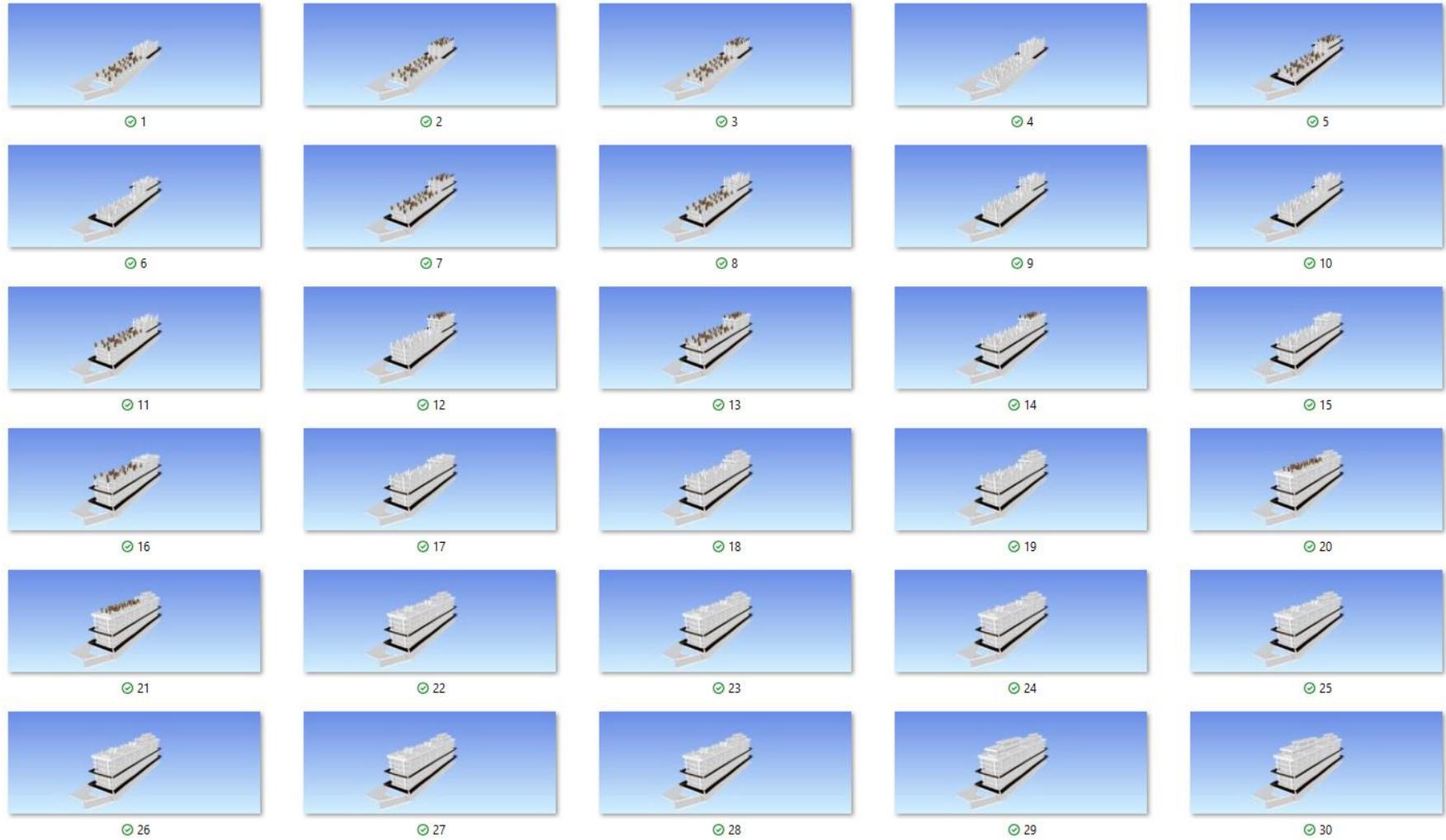


Figura 52: Simulação 15

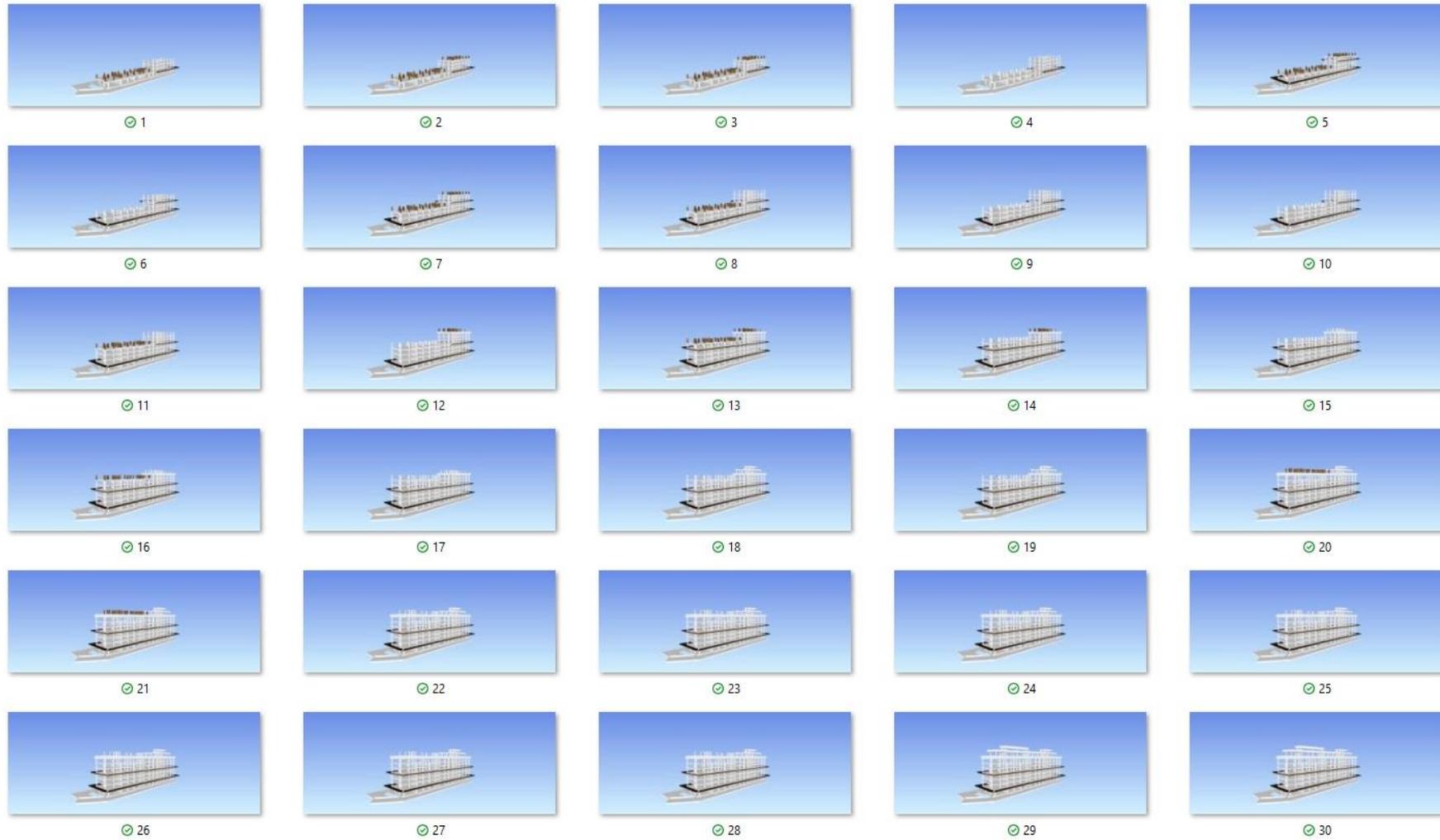


Figura 53: Simulação 16

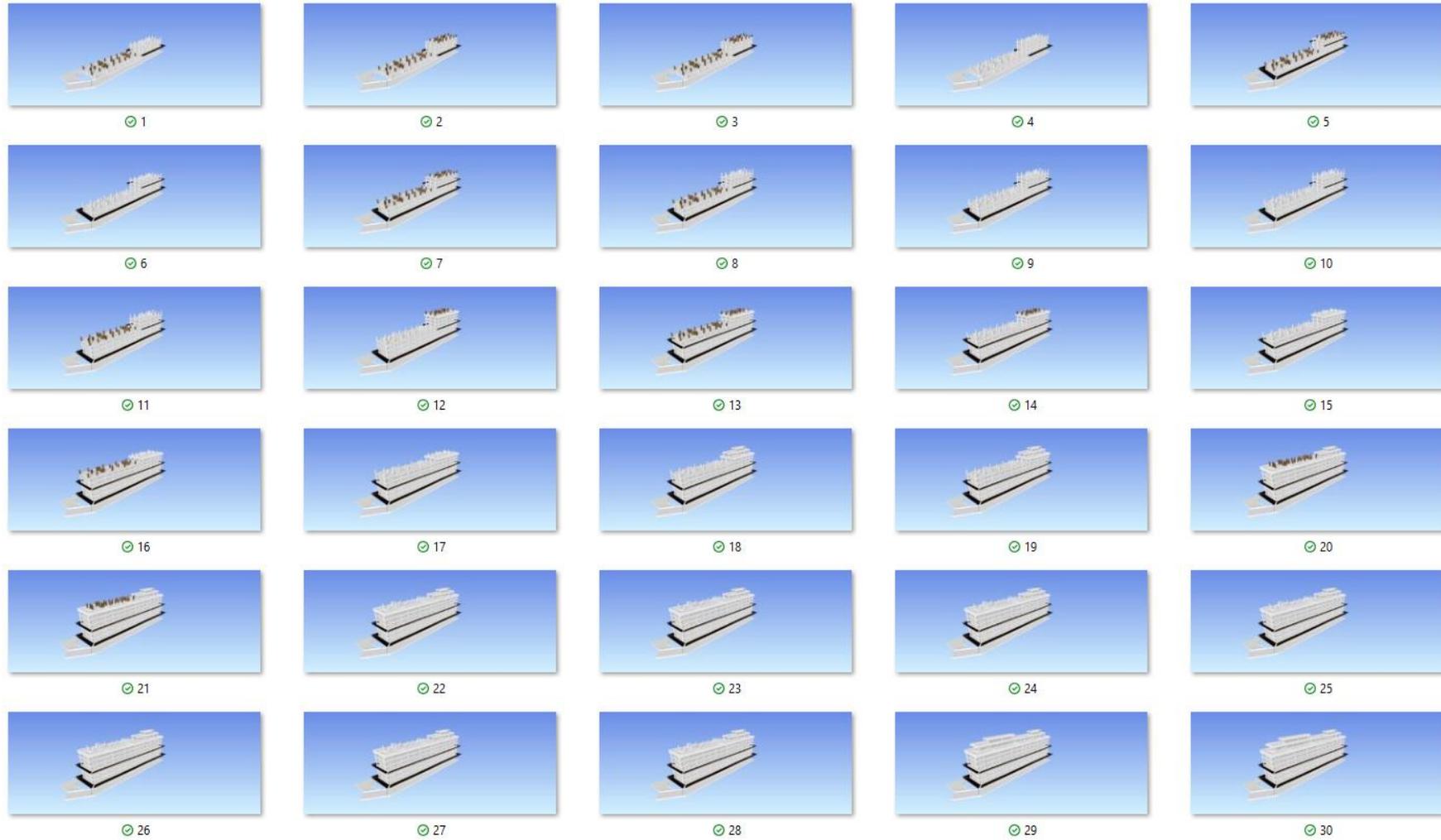


Figura 54: Simulação 17

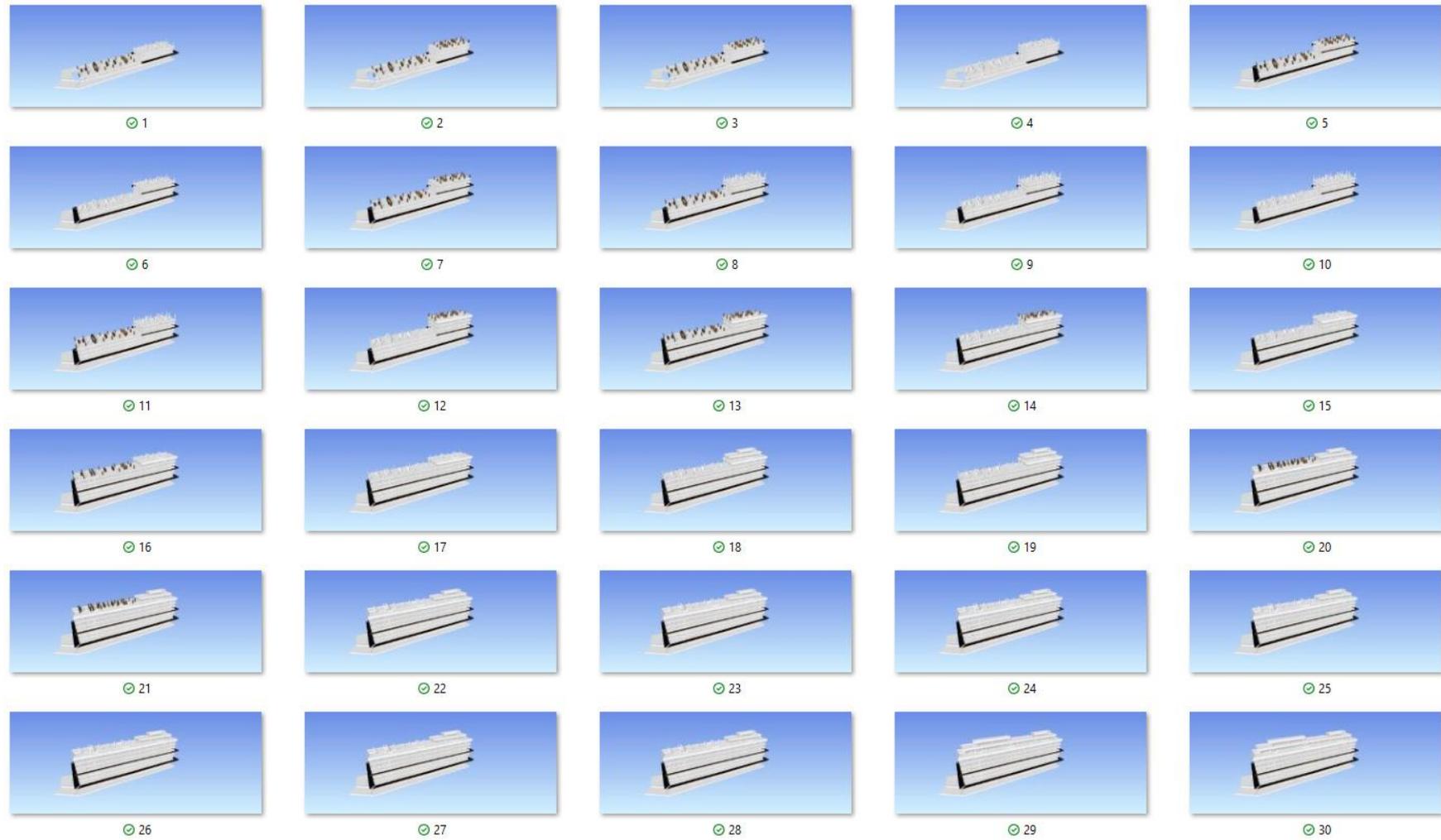


Figura 55: Simulação 18

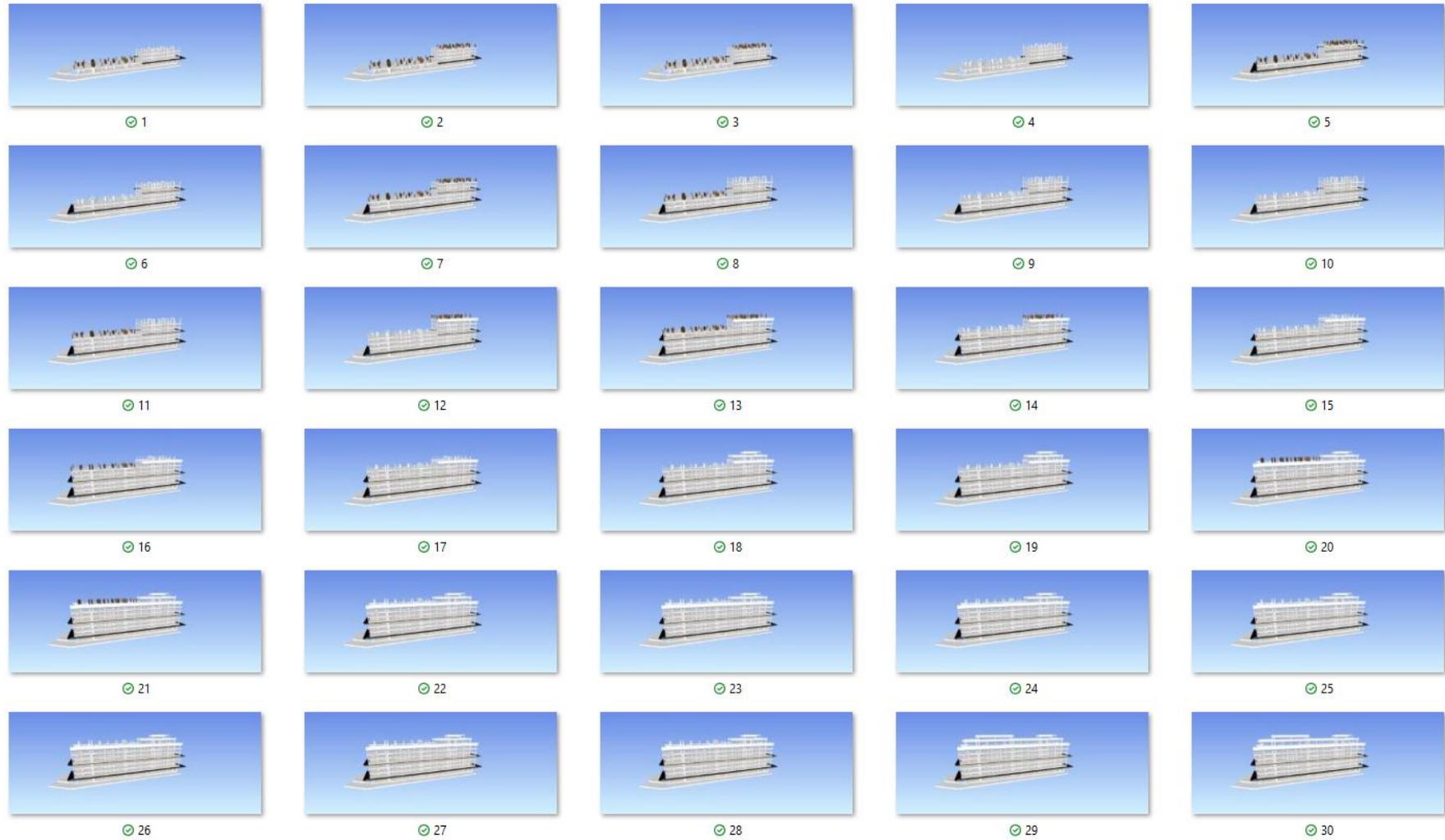


Figura 56: Simulação 19

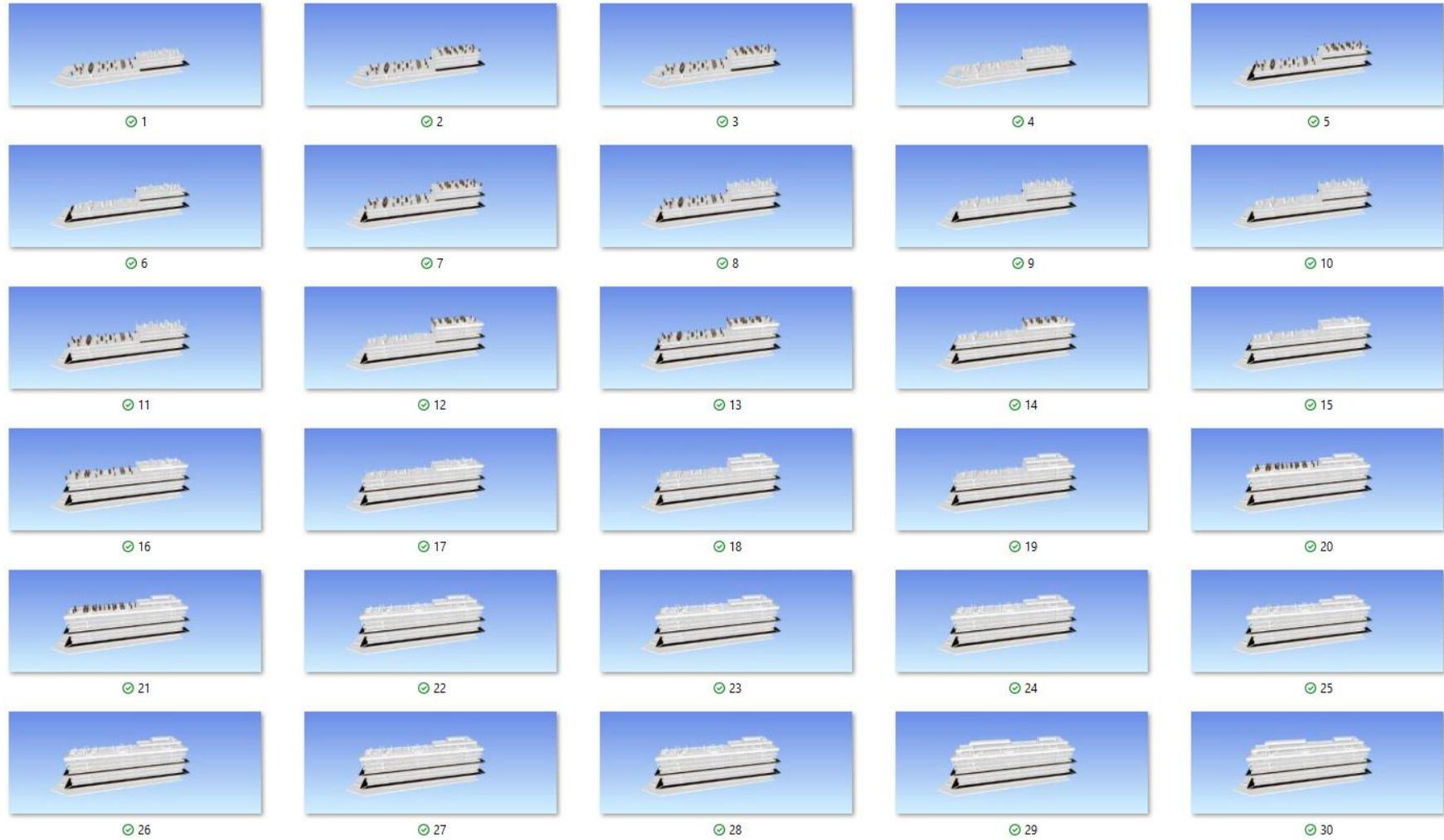


Figura 57: Simulação 20



Figura 58: Simulação 21

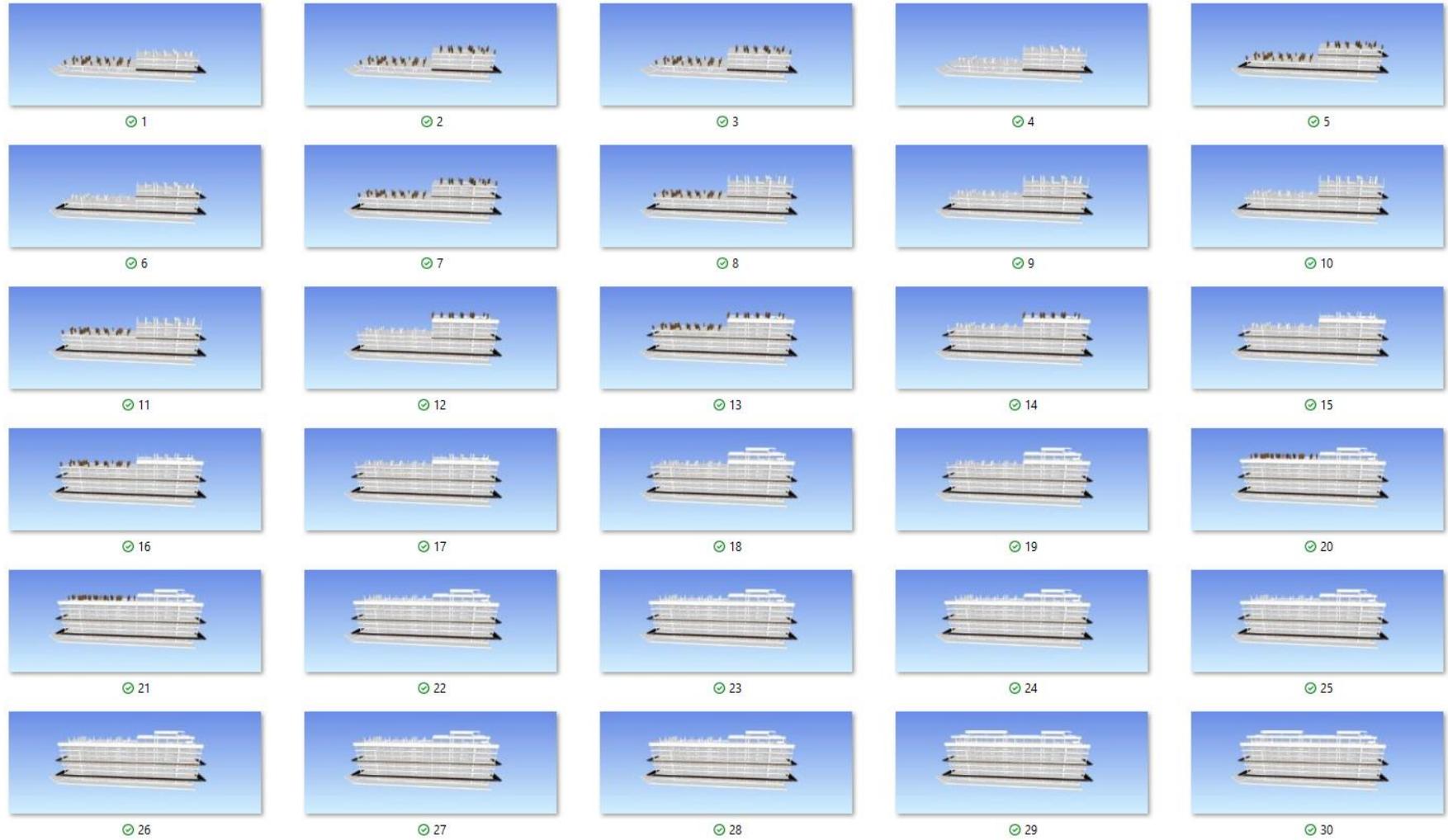


Figura 59: Simulação 22

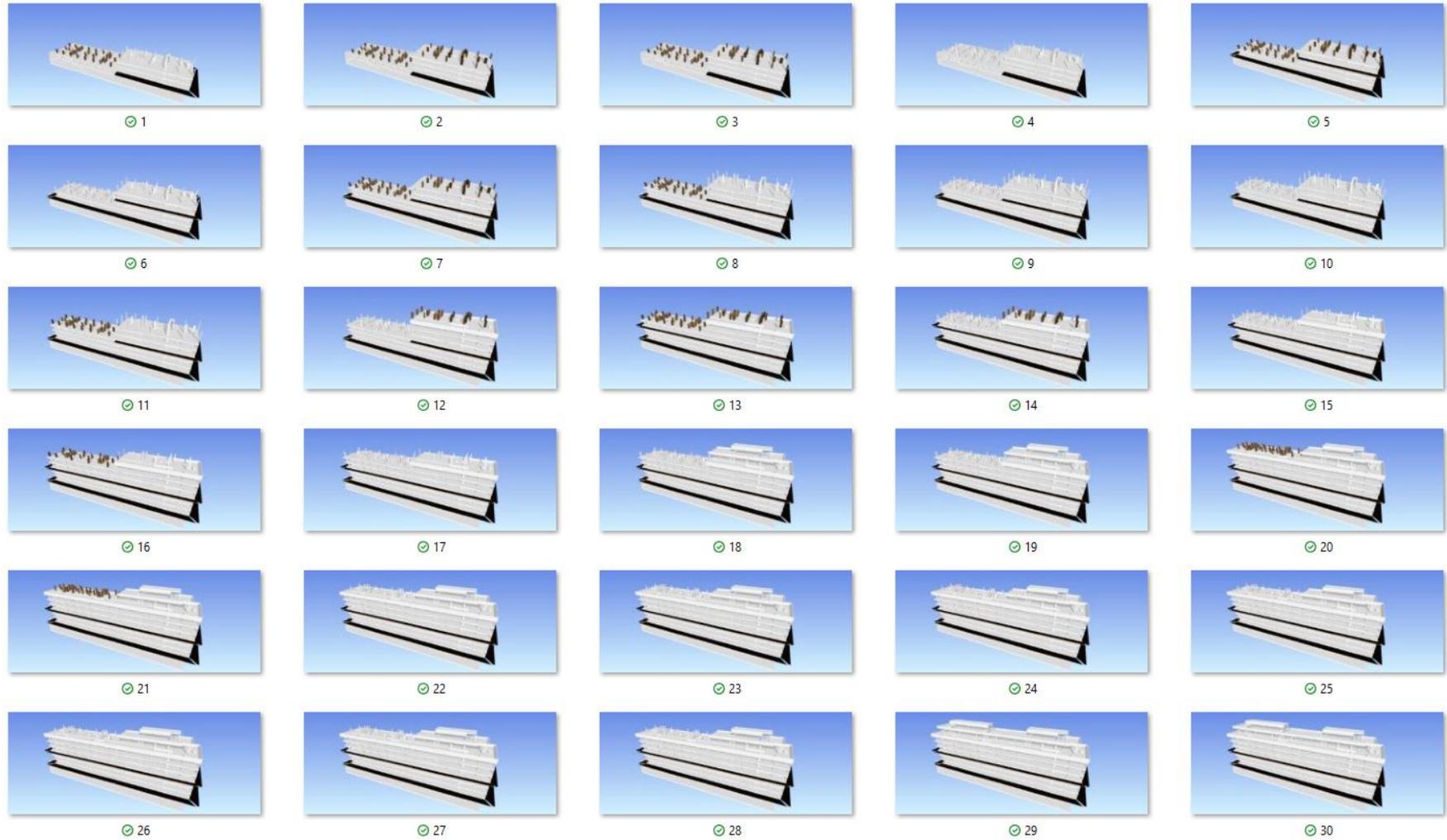


Figura 60: Simulação 23

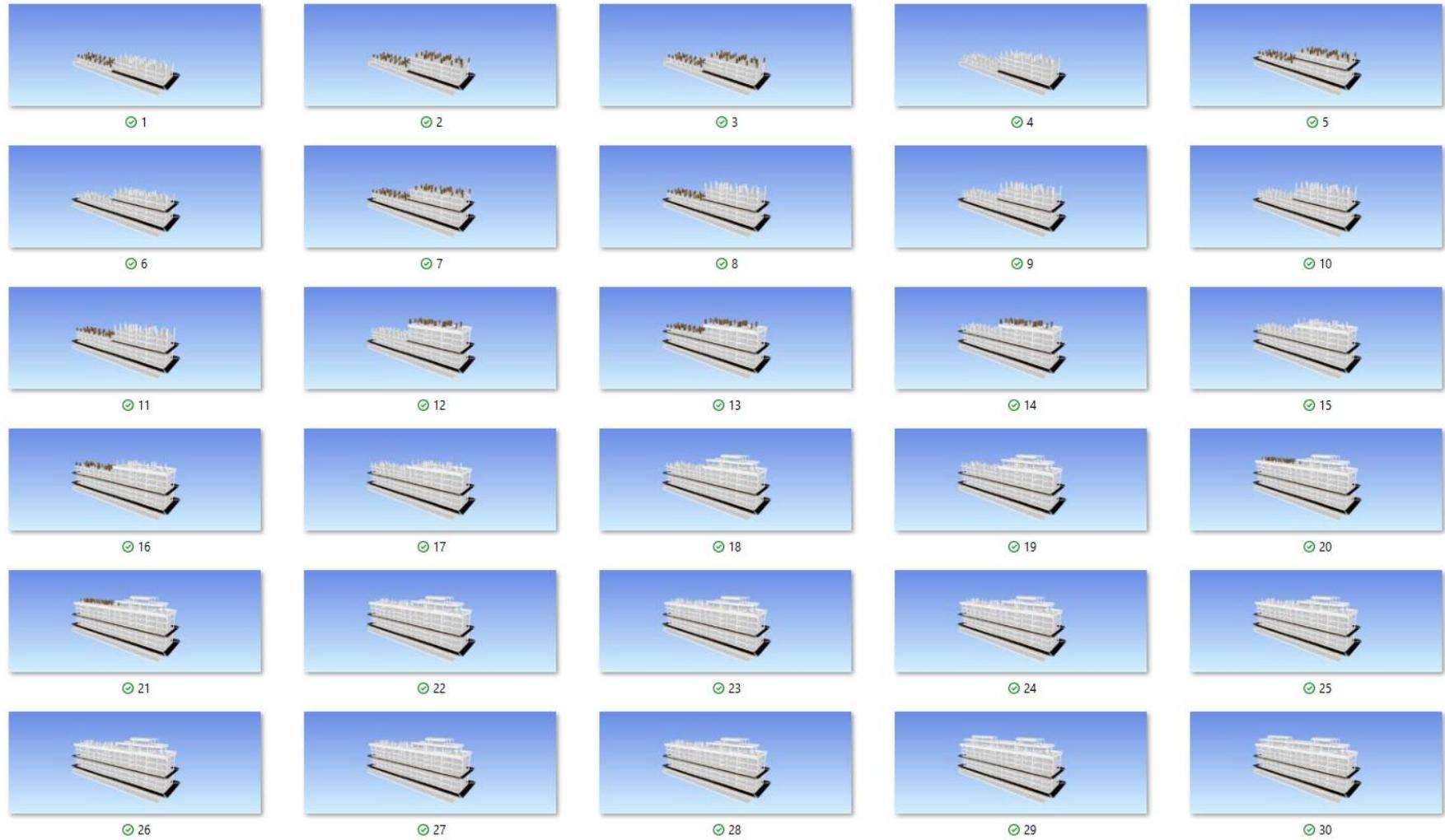


Figura 61: Simulação 24

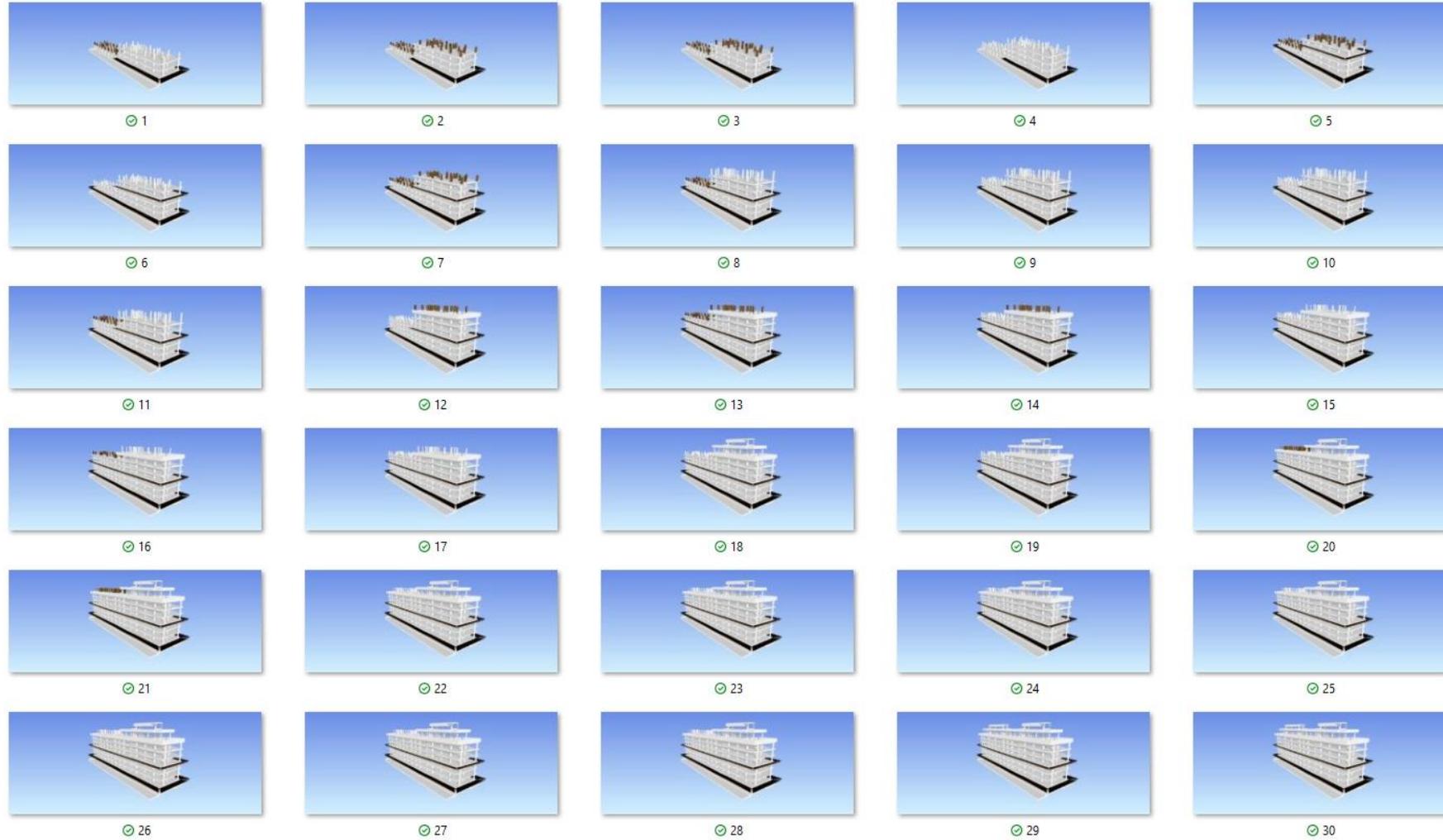


Figura 62: Simulação 25

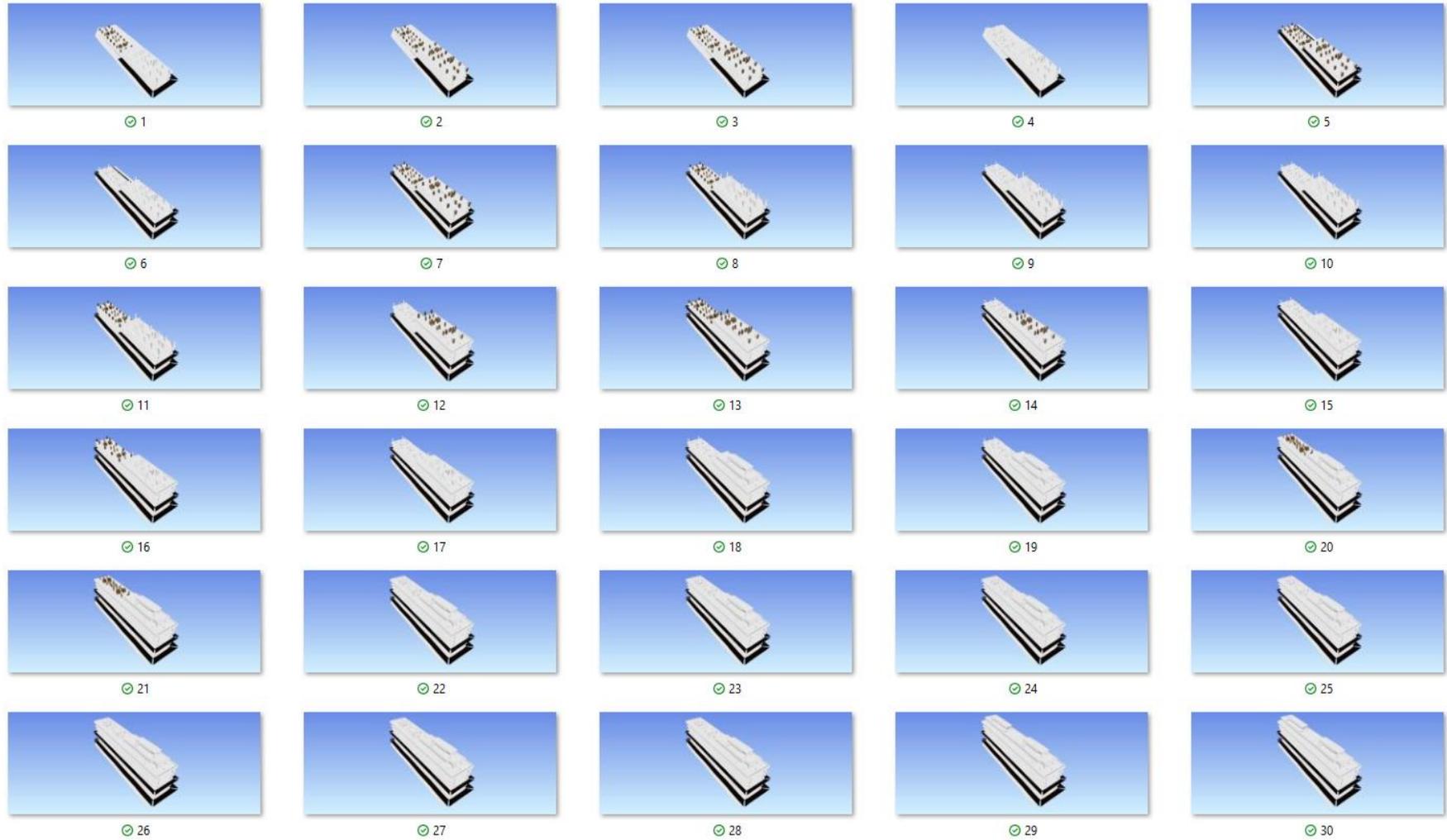


Figura 63: Simulação 26

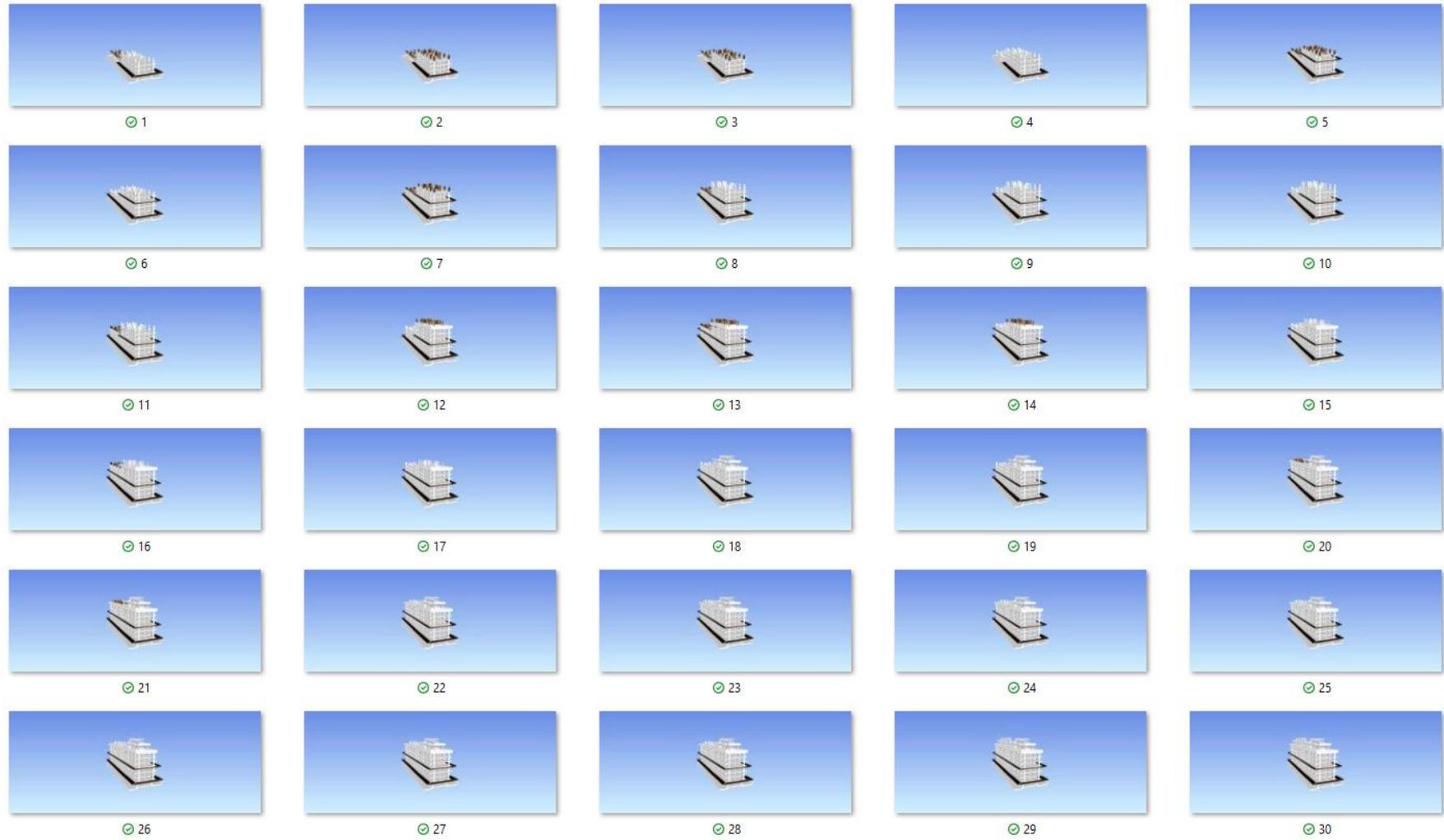


Figura 64: Simulação 27

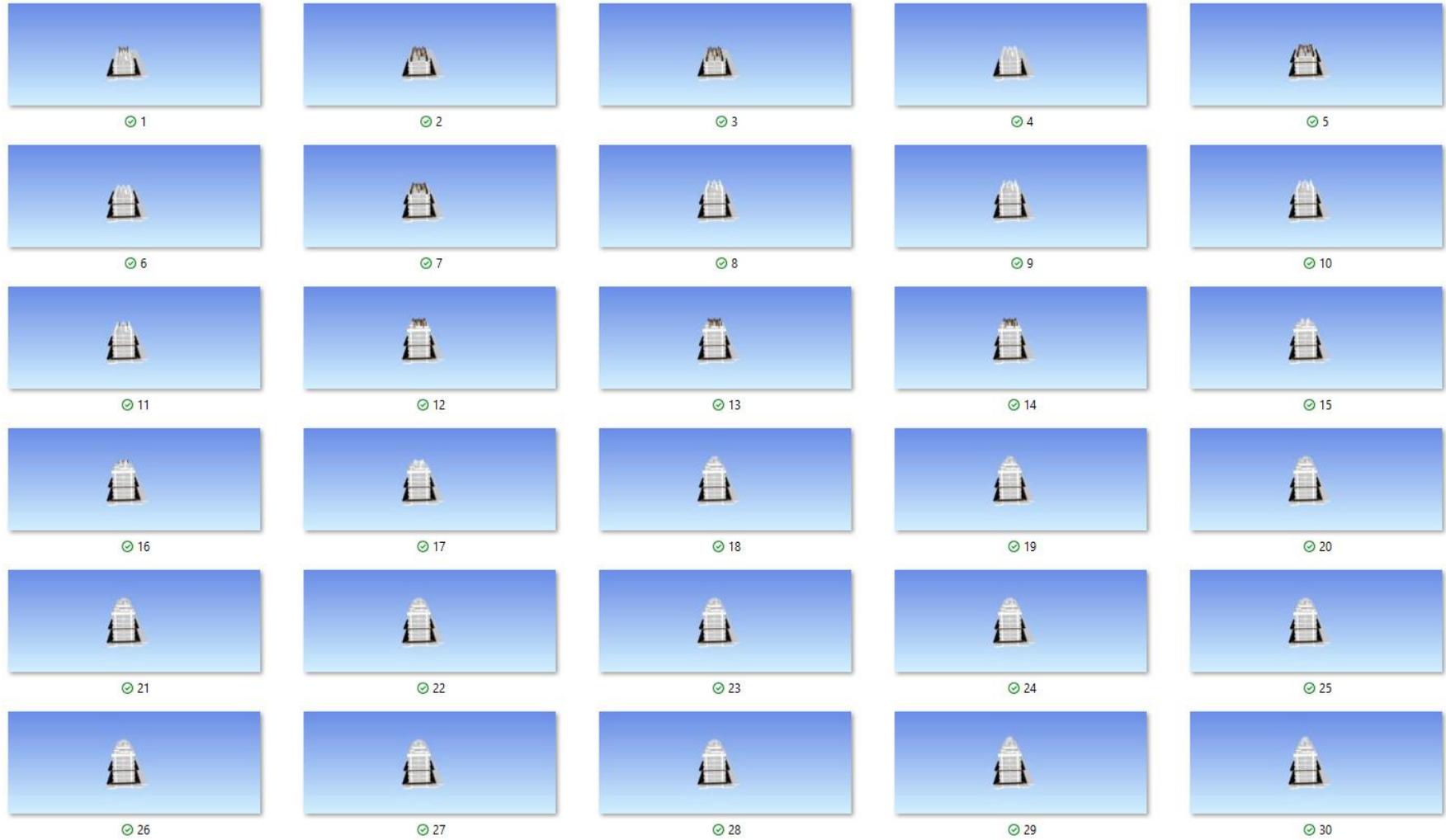


Figura 65: Simulação 28

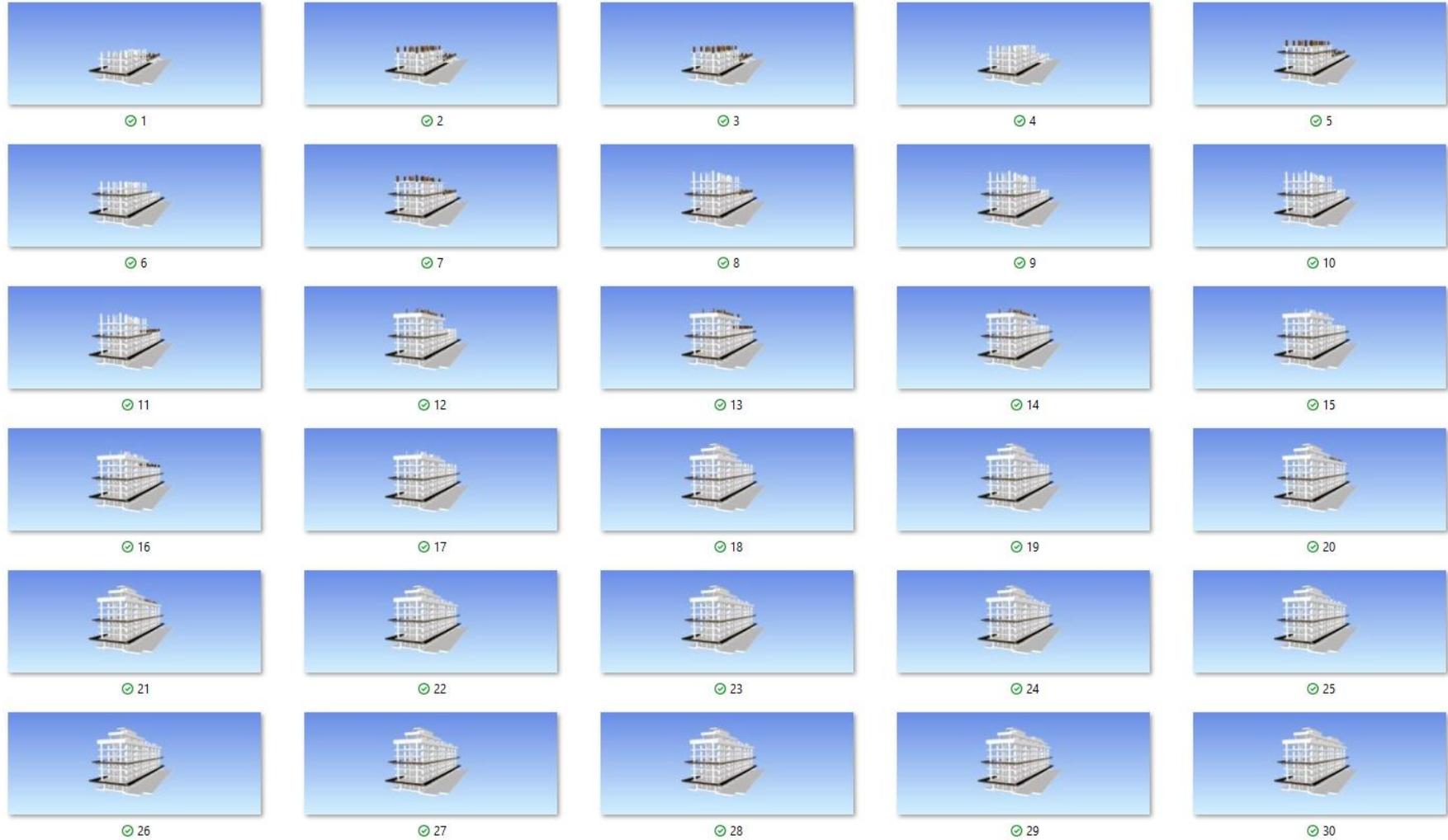


Figura 66: Simulação 29

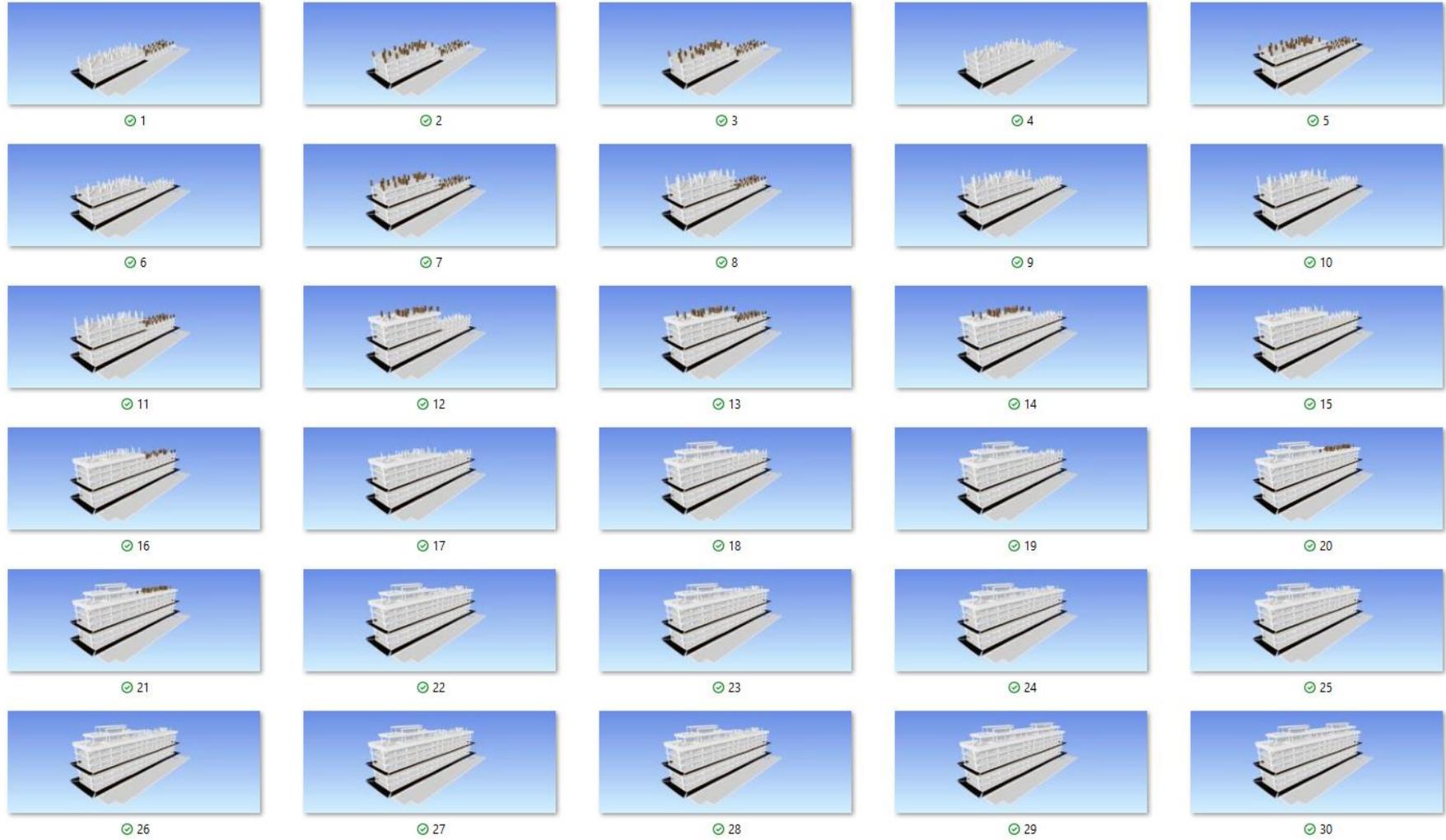


Figura 67: Simulação 30

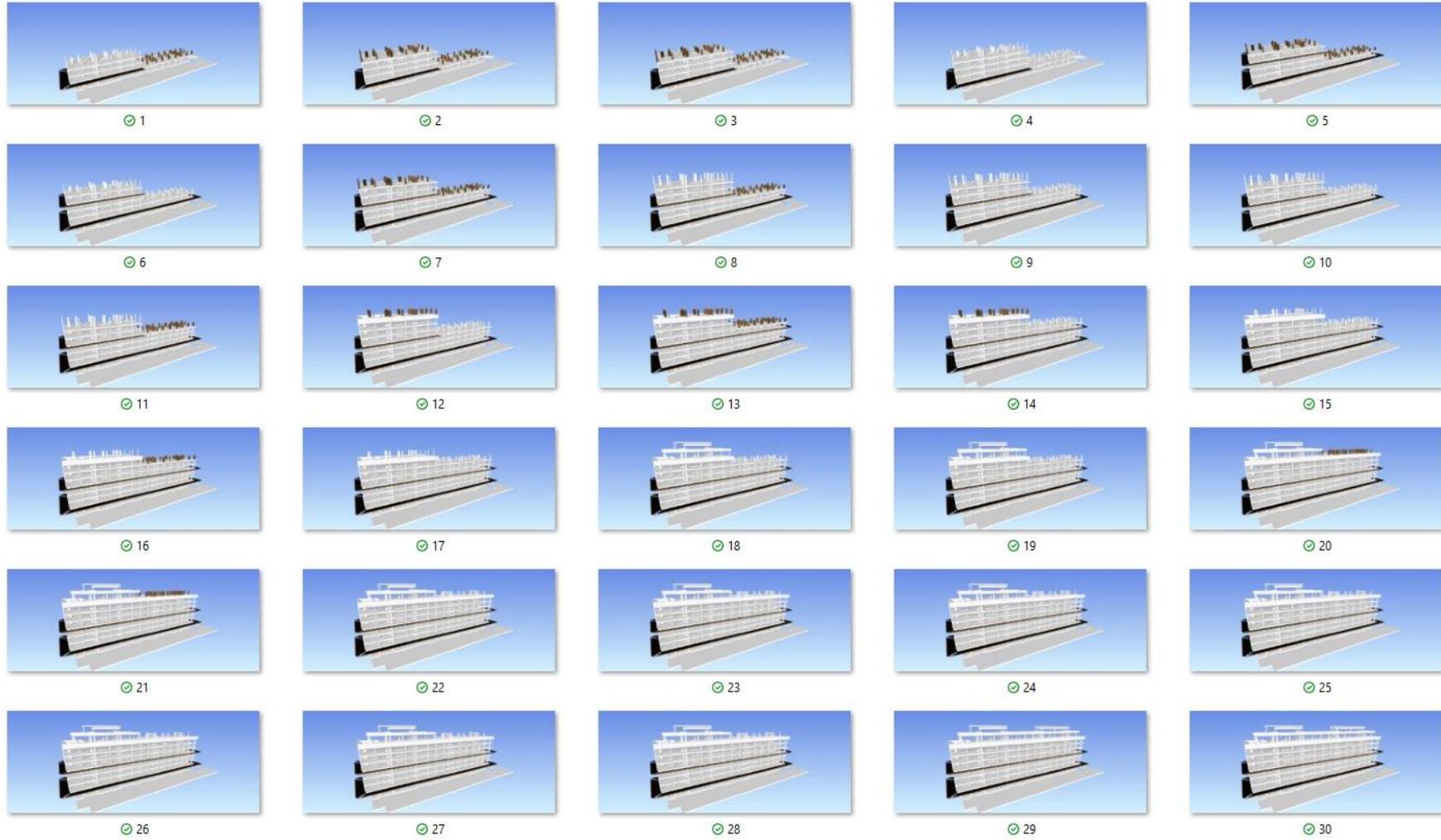


Figura 68: Simulação 31

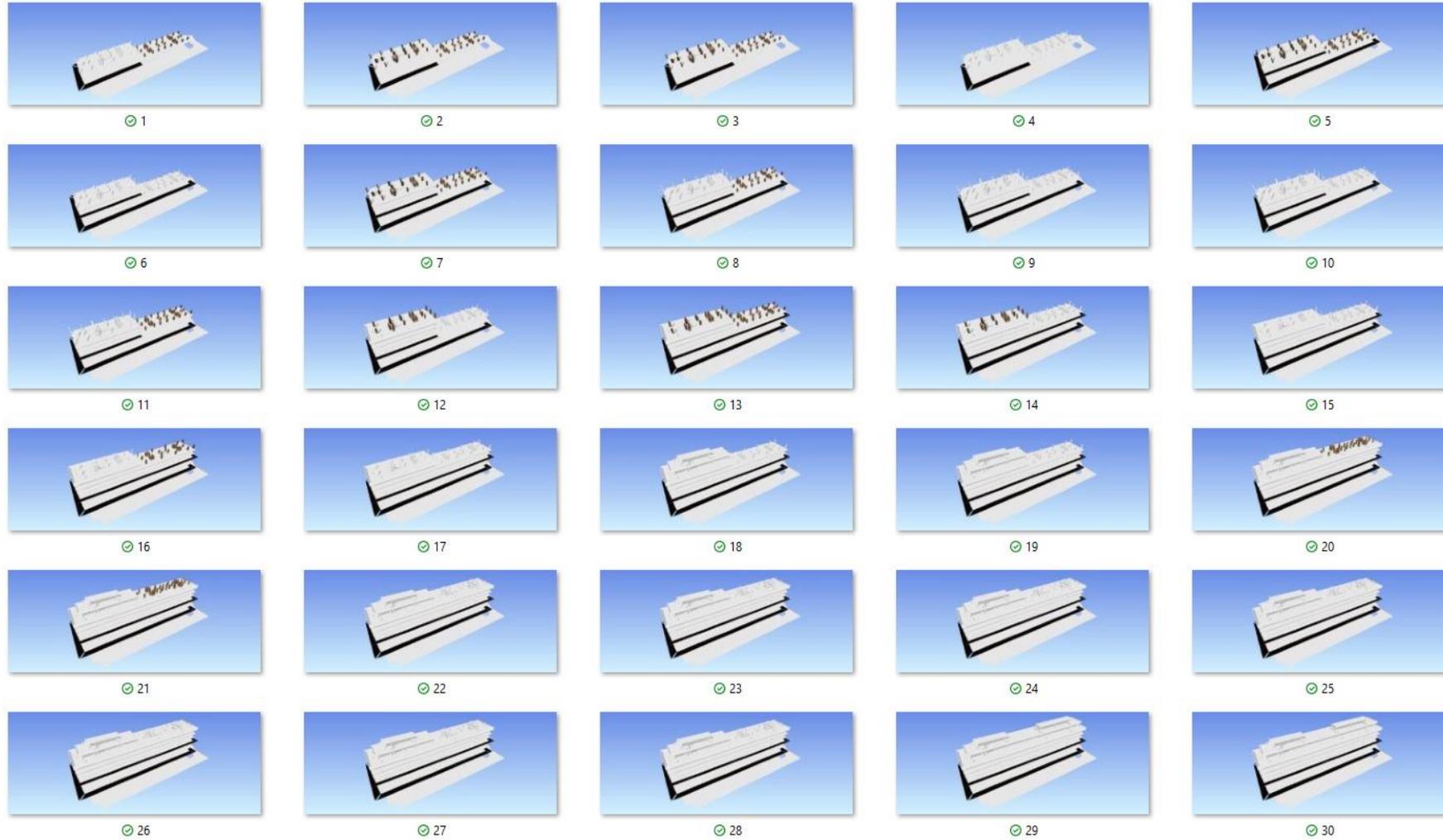


Figura 69: Simulação 32

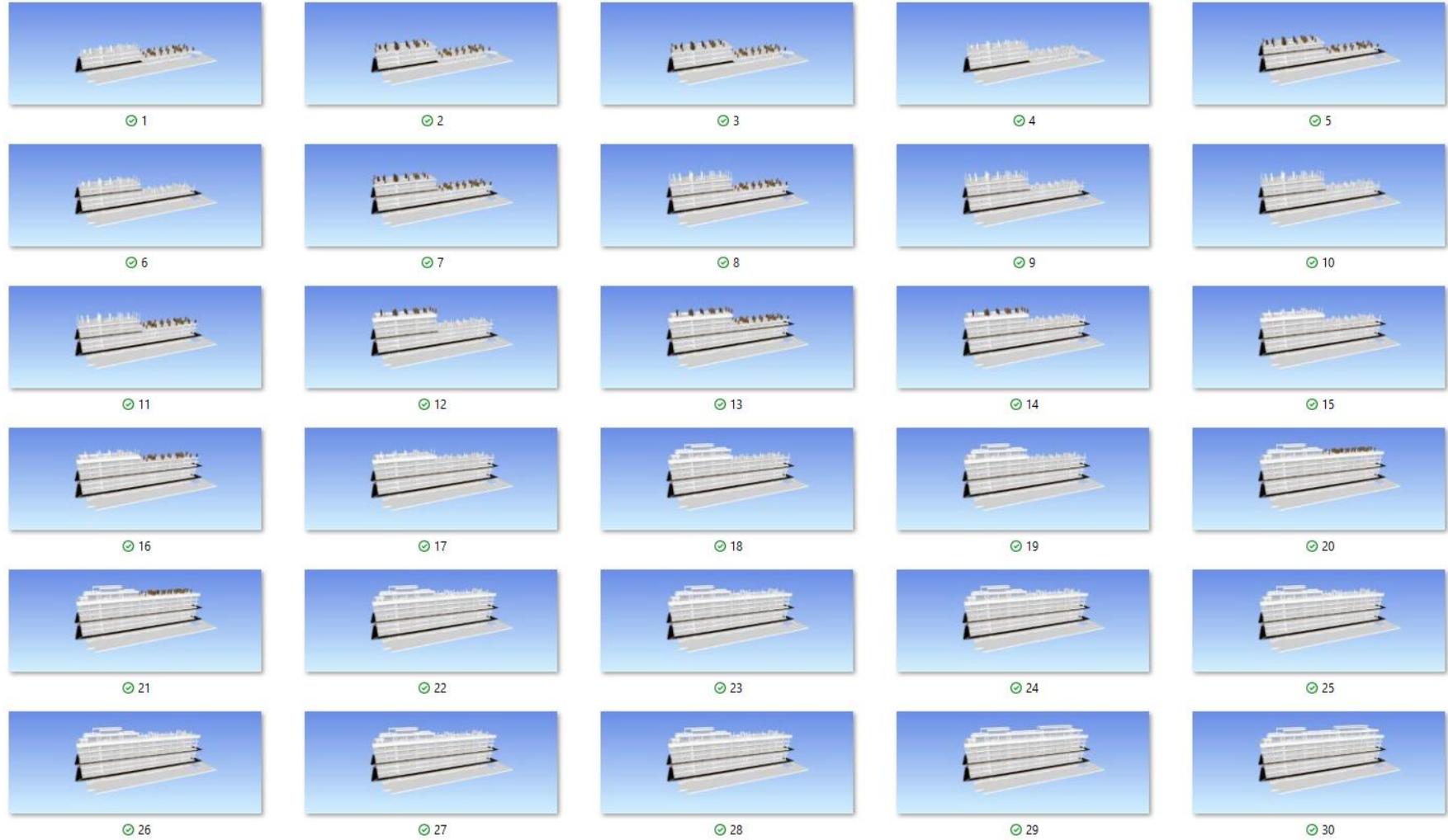


Figura 70: Simulação 33

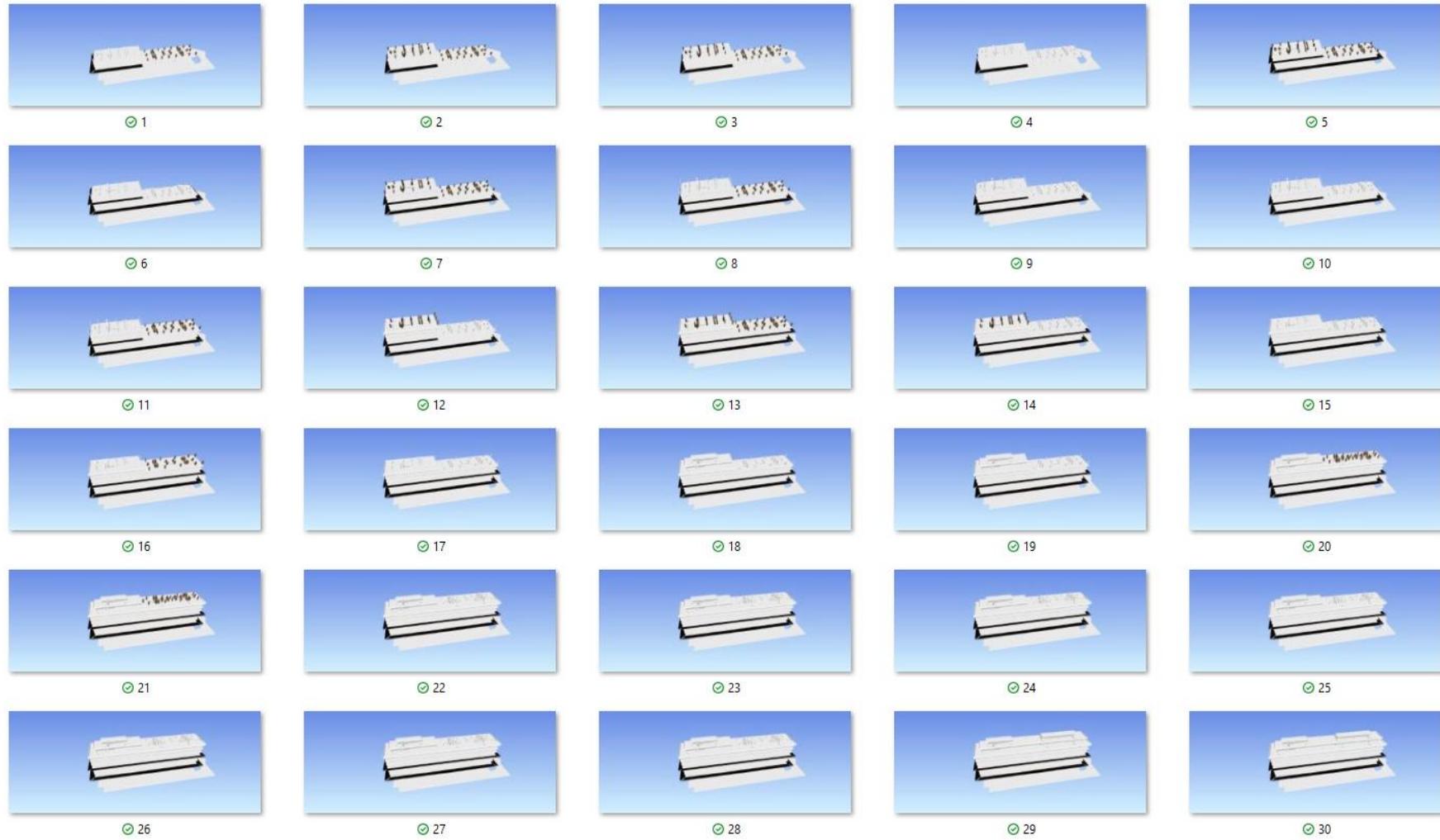


Figura 71: Simulação 34

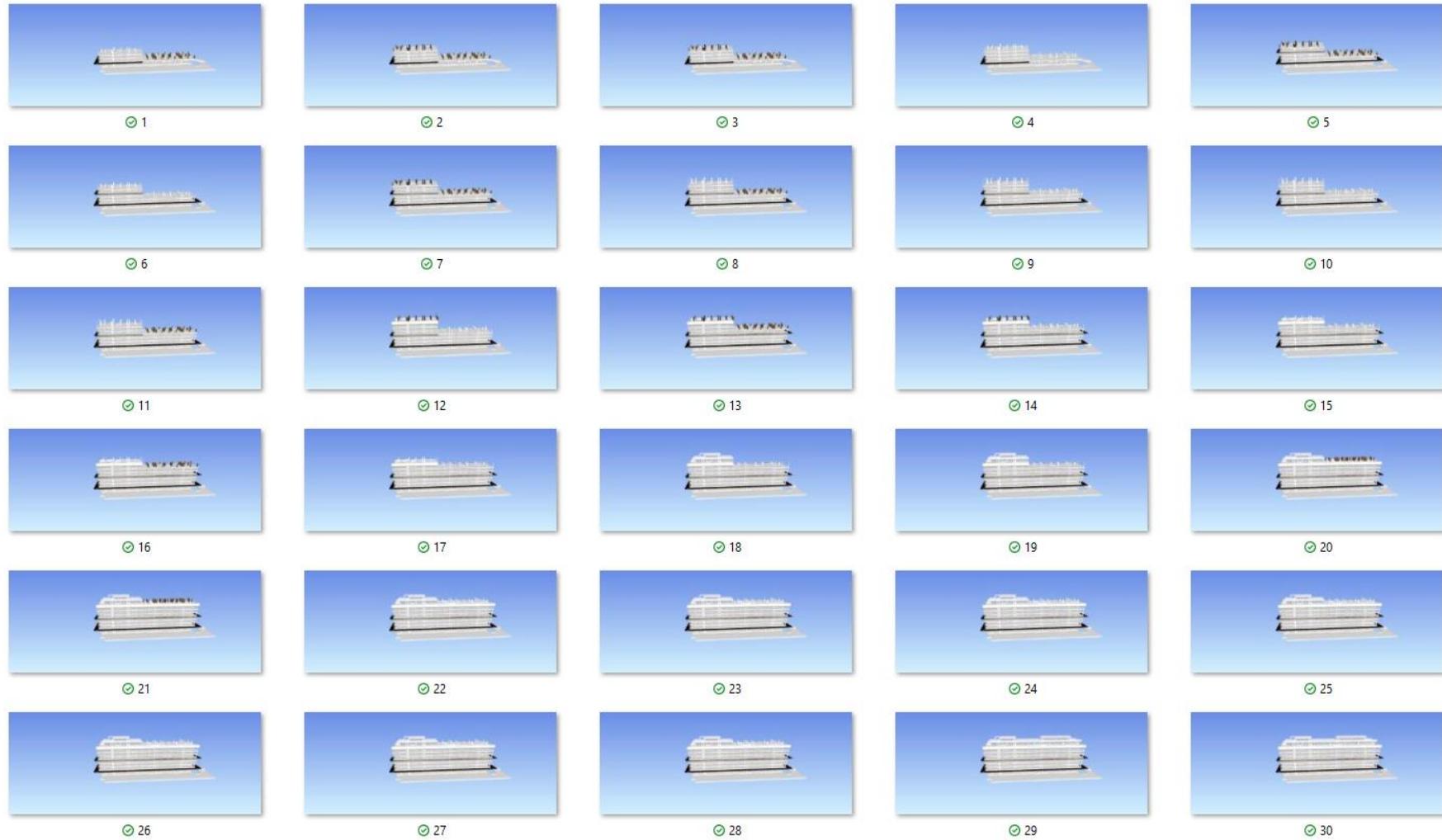


Figura 72: Simulação 35



Figura 73: Simulação 36

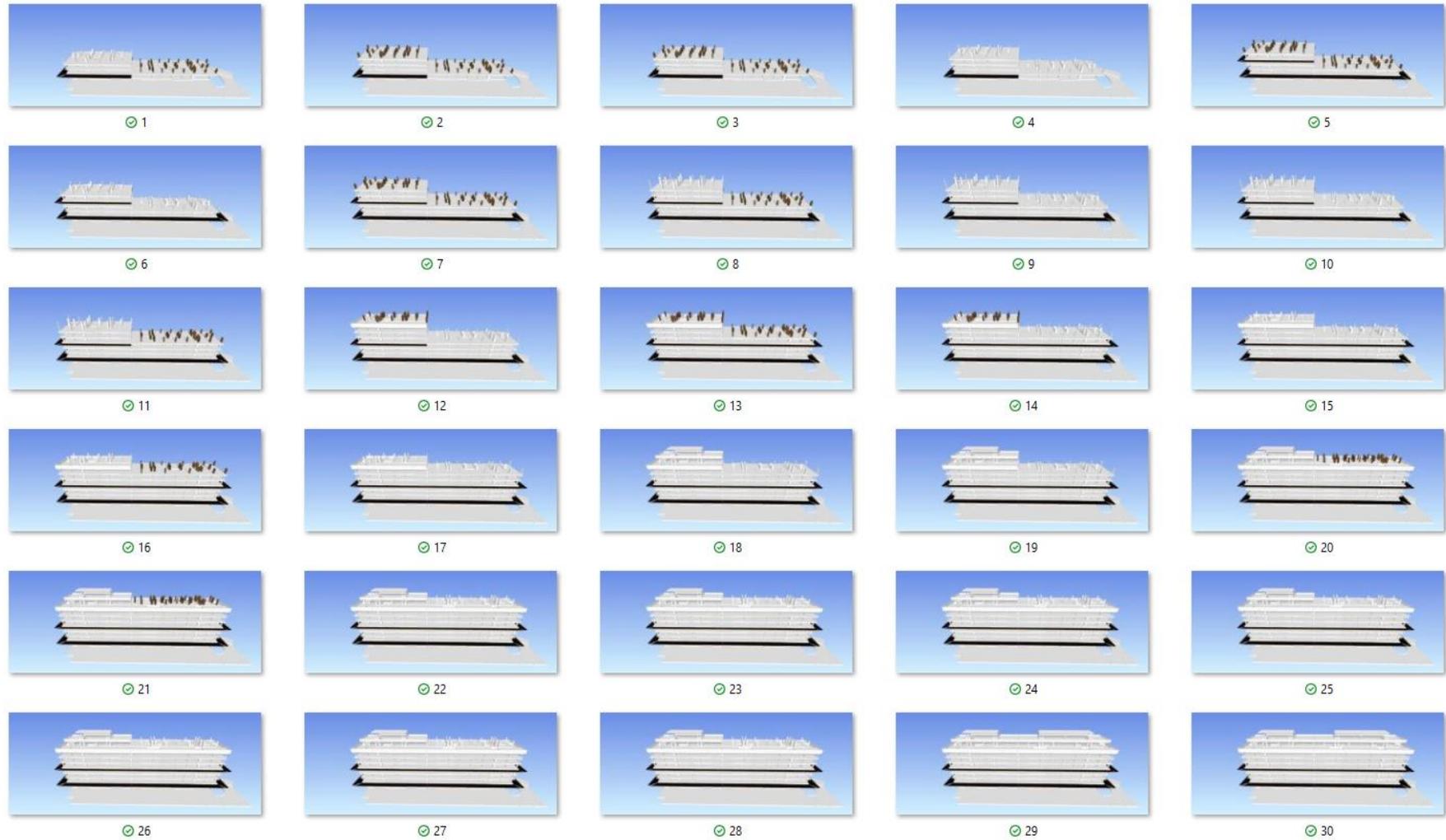


Figura 74: Simulação 37

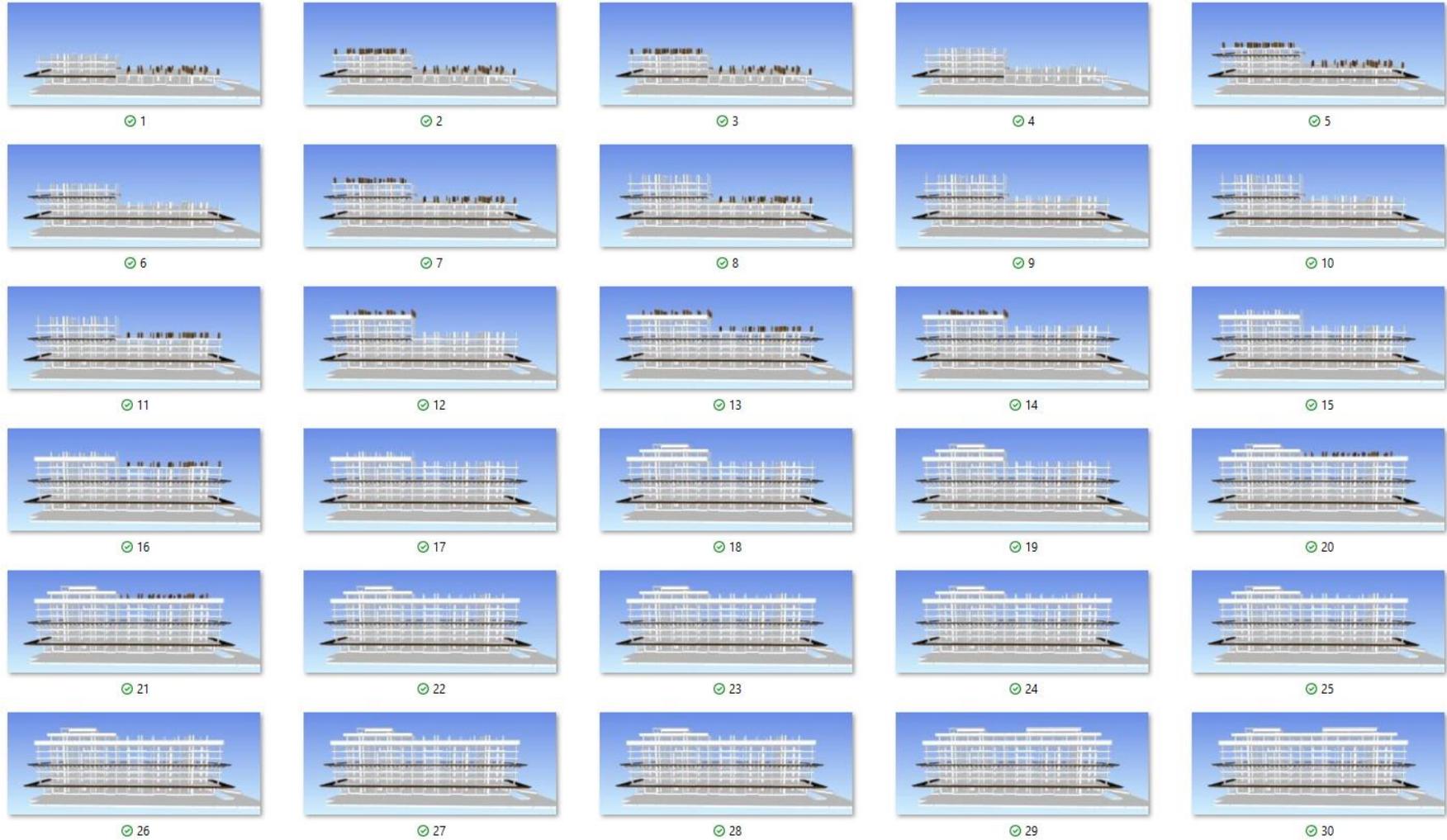


Figura 75: Simulação 38

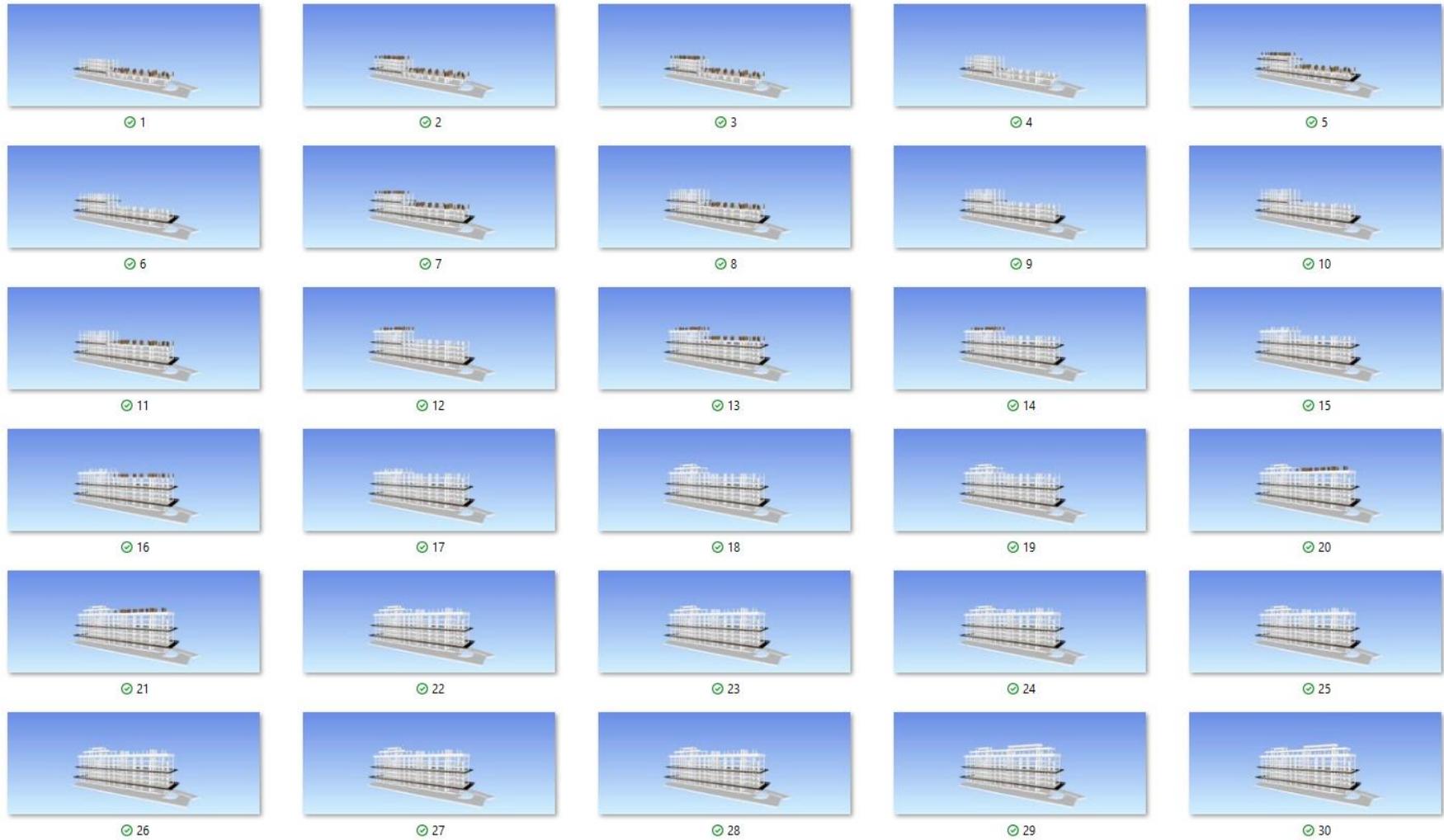
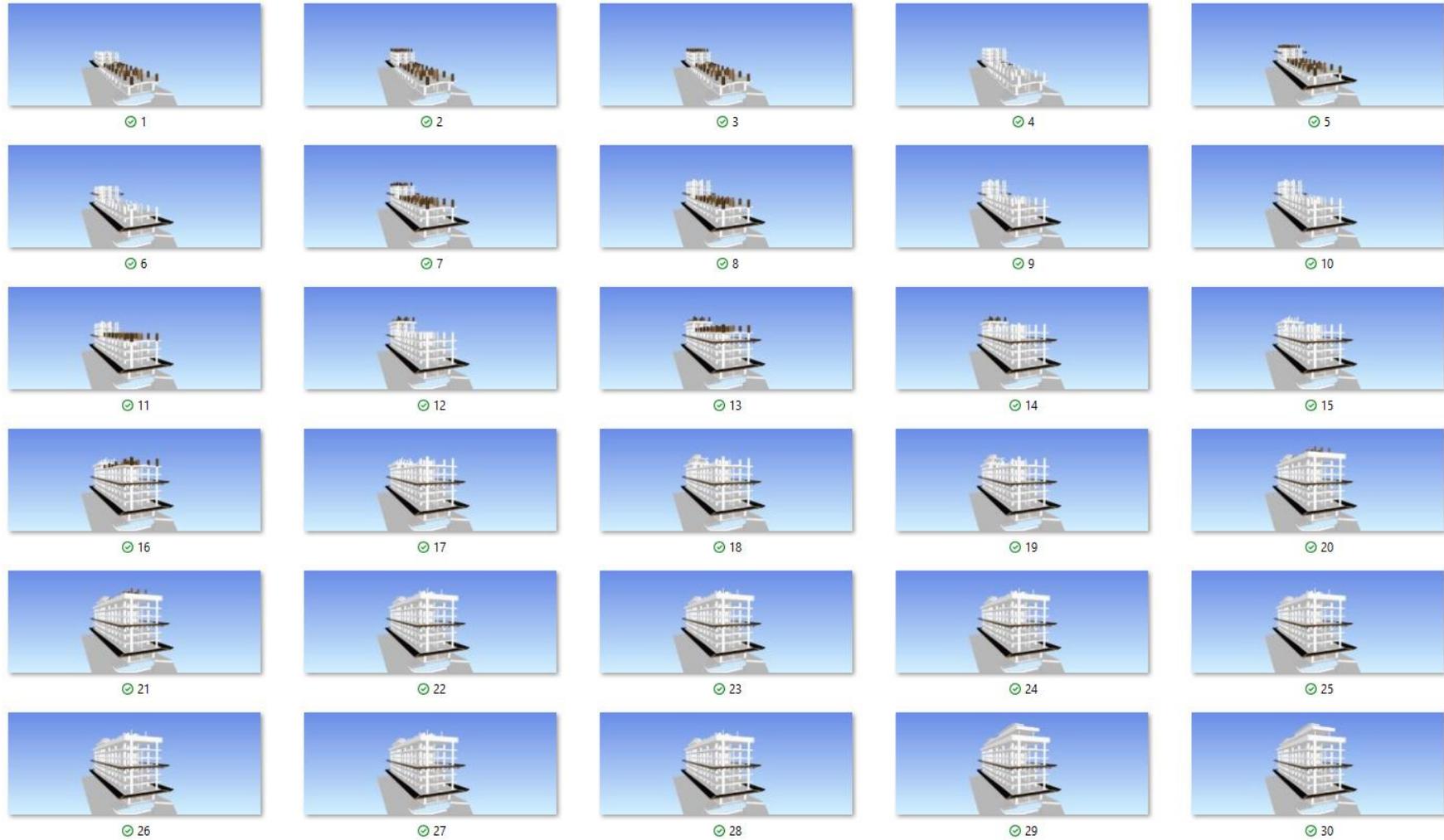


Figura 76: Simulação 39



Figura 77: Simulação 40



APÊNDICE II

```
#Importando as bibliotecas necessárias
import os, os.path

#Renomeando as imagens
valid_images = [".jpg", ".jpeg"]
for j in range (1,41):
    path = str("Simulacao " + str(j))
    #path = "Simulacao 2"
    i = 1
    for f in os.listdir(path):
        ext = os.path.splitext(f)[1]
        f_name = os.path.splitext(f)[0]
        if ext.lower() not in valid_images:
            continue
        if f_name == "Simulação Canteiro-TimeLiner-053":
            os.rename(os.path.join(path,f), os.path.join(path,"1.jpg"))
        elif f_name == "Simulação Canteiro-TimeLiner-054":
            os.rename(os.path.join(path,f), os.path.join(path,"2.jpg"))
        elif f_name == "Simulação Canteiro-TimeLiner-055":
            os.rename(os.path.join(path,f), os.path.join(path,"3.jpg"))
        elif f_name == "Simulação Canteiro-TimeLiner-056":
            os.rename(os.path.join(path,f), os.path.join(path,"4.jpg"))
        elif f_name == "Simulação Canteiro-TimeLiner-057":
            os.rename(os.path.join(path,f), os.path.join(path,"5.jpg"))
        elif f_name == "Simulação Canteiro-TimeLiner-058":
            os.rename(os.path.join(path,f), os.path.join(path,"6.jpg"))
        elif f_name == "Simulação Canteiro-TimeLiner-059":
            os.rename(os.path.join(path,f), os.path.join(path,"7.jpg"))
        elif f_name == "Simulação Canteiro-TimeLiner-060":
            os.rename(os.path.join(path,f), os.path.join(path,"8.jpg"))
        elif f_name == "Simulação Canteiro-TimeLiner-061":
            os.rename(os.path.join(path,f), os.path.join(path,"9.jpg"))
        elif f_name == "Simulação Canteiro-TimeLiner-062":
```

```
    os.rename(os.path.join(path,f), os.path.join(path,"10.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-063":
    os.rename(os.path.join(path,f), os.path.join(path,"11.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-064":
    os.rename(os.path.join(path,f), os.path.join(path,"12.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-065":
    os.rename(os.path.join(path,f), os.path.join(path,"13.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-066":
    os.rename(os.path.join(path,f), os.path.join(path,"14.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-067":
    os.rename(os.path.join(path,f), os.path.join(path,"15.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-068":
    os.rename(os.path.join(path,f), os.path.join(path,"16.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-069":
    os.rename(os.path.join(path,f), os.path.join(path,"17.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-070":
    os.rename(os.path.join(path,f), os.path.join(path,"18.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-071":
    os.rename(os.path.join(path,f), os.path.join(path,"19.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-072":
    os.rename(os.path.join(path,f), os.path.join(path,"20.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-073":
    os.rename(os.path.join(path,f), os.path.join(path,"21.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-074":
    os.rename(os.path.join(path,f), os.path.join(path,"22.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-075":
    os.rename(os.path.join(path,f), os.path.join(path,"23.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-076":
    os.rename(os.path.join(path,f), os.path.join(path,"24.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-077":
    os.rename(os.path.join(path,f), os.path.join(path,"25.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-078":
    os.rename(os.path.join(path,f), os.path.join(path,"26.jpg"))
elif f_name == "Simulação Canteiro-TimeLiner-079":
    os.rename(os.path.join(path,f), os.path.join(path,"27.jpg"))
```

```
elif f_name == "Simulação Canteiro-TimeLiner-080":  
    os.rename(os.path.join(path,f), os.path.join(path,"28.jpg"))  
elif f_name == "Simulação Canteiro-TimeLiner-081":  
    os.rename(os.path.join(path,f), os.path.join(path,"29.jpg"))  
elif f_name == "Simulação Canteiro-TimeLiner-082":  
    os.rename(os.path.join(path,f), os.path.join(path,"30.jpg"))
```

APÊNDICE III

```
#Importação das bibliotecas necessárias
from PIL import Image
import os, os.path
import cv2
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

#Inserção das imagens geradas em uma lista do Python
imgs = [1]*40*30
valid_images = [".jpg", ".jpeg"]
i=0
for j in range(1,41):
    path = "Simulacao " + str(j)
    for f in os.listdir(path):
        ext = os.path.splitext(f)[1]
        f_name = os.path.splitext(f)[0]
        if ext.lower() not in valid_images:
            continue
        imgs[(j-1)*30 + int(f_name)-1] = cv2.imread(os.path.join(path,f))
        imgs[(j-1)*30 + int(f_name)-1] = cv2.cvtColor(imgs[(j-1)*30 + int(f_name)-1],
cv2.COLOR_BGR2RGB)
        imgs[(j-1)*30 + int(f_name)-1] = cv2.resize(imgs[(j-1)*30 + int(f_name)-1],(256,256))
        imgs[(j-1)*30 + int(f_name)-1] = imgs[(j-1)*30 + int(f_name)-1]/255
        i = i+1

#Verificando se as imagens foram inseridas na lista corretamente
plt.imshow(imgs[48])
type(imgs[10])
for i in range(1,901):
    print(str(i) + "-" + str(type(imgs[i])))

#Transformando a lista de imagens em um vetor do NumPy
```

```
X = np.array(imgs)

#Inserção do progresso da obra em cada imagem como um DataFrame
y = pd.read_excel("Quantitativos.xlsx")

#Divisão do banco de dados em treinamento e teste
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X,y, test_size = 0.25,
                                                    random_state = 42)

X_train = pd.DataFrame.to_csv("X_train.csv", encoding = 'utf-8', index = True)
X_test = pd.DataFrame.to_csv("X_test.csv", encoding = 'utf-8', index = True)
y_train = pd.DataFrame.to_csv("y_train.csv", encoding = 'utf-8', index = True)
y_test = pd.DataFrame.to_csv("y_test.csv", encoding = 'utf-8', index = True)
```

APÊNDICE IV

Figura 78: Estrutura da Rede Neural Convolutacional 1 (RNC_1)

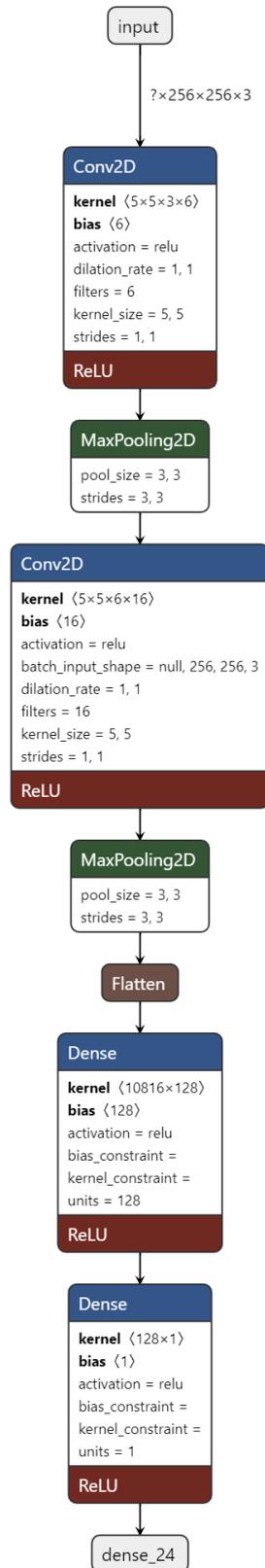


Figura 79: Estrutura da Rede Neural Convolutacional 2 (RNC_2)

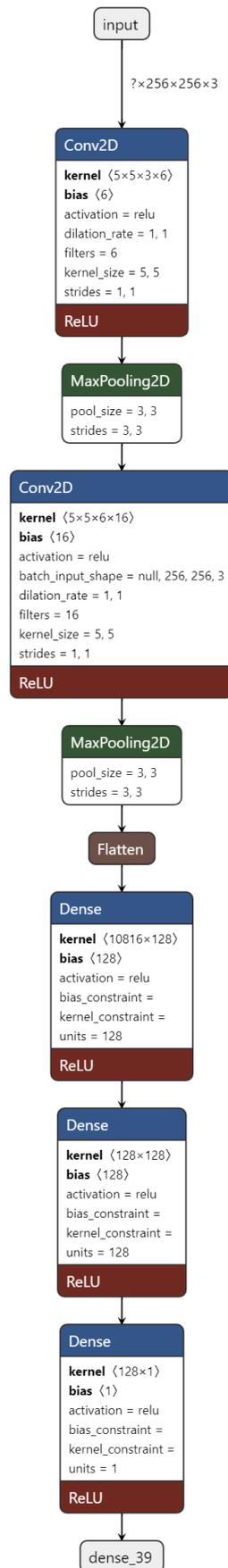
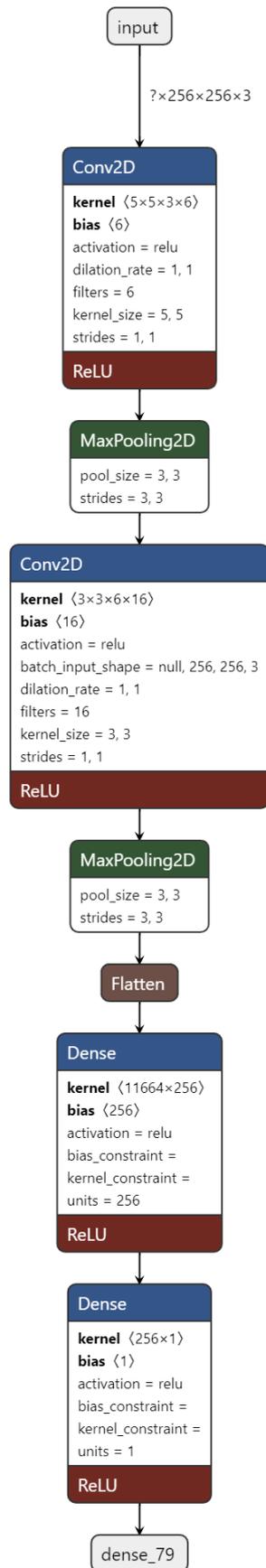


Figura 80: Estrutura da Rede Neural Convolutacional 3 (RNC_3)



APÊNDICE V

```
#Criando a estrutura e treinando as redes neurais convolucionais=====
#Importando as bibliotecas necessárias
from keras.models import Sequential
from keras.layers import Dense,Conv2D,MaxPool2D, Flatten

#RNC1=====
#Criando a estrutura da RNC 1
CNN_1 = Sequential()
CNN_1.add(Conv2D(filters=6,
                 kernel_size=5,activation="relu",
                 input_shape=(256,256,3)))
CNN_1.add(MaxPool2D(pool_size=3,
                   padding="valid"))
CNN_1.add(Conv2D(filters=16,
                 kernel_size=5,activation="relu",
                 input_shape=(256,256,3)))
CNN_1.add(MaxPool2D(pool_size=3,
                   padding="valid"))
CNN_1.add(Flatten())
CNN_1.add(Dense(128,activation="relu"))
CNN_1.add(Dense(1,activation="relu"))
CNN_1.compile(loss='mean_squared_error',
              metrics=['mean_absolute_error'],
              optimizer = 'adam')
CNN_1.summary()

#Treinando a RNC 1
CNN_1.fit(X_train,y_train,batch_size = 5, epochs = 100)

#Salvando a RNC 1
CNN_1.save('CNN_1.h5')

#Carregando a RNC 1
```

```

from tensorflow import keras
CNN_1 = keras.models.load_model('CNN_1.h5')

#RNC2=====
#Criando a estrutura da RNC 2
CNN_2 = Sequential()
CNN_2.add(Conv2D(filters=6,
                 kernel_size=5,activation="relu",
                 input_shape=(256,256,3)))
CNN_2.add(MaxPool2D(pool_size=3,
                   padding="valid"))
CNN_2.add(Conv2D(filters=16,
                 kernel_size=5,activation="relu",
                 input_shape=(256,256,3)))
CNN_2.add(MaxPool2D(pool_size=3,
                   padding="valid"))
CNN_2.add(Flatten())
CNN_2.add(Dense(128,activation="relu"))
CNN_2.add(Dense(128,activation="relu"))
CNN_2.add(Dense(1,activation="relu"))
CNN_2.compile(loss='mean_squared_error',
              metrics=['mean_absolute_error'],
              optimizer = 'adam')
CNN_2.summary()

#Treinando a RNC 2
CNN_2.fit(X_train,y_train,batch_size = 5, epochs = 100)

#Salvando a RNC 2
CNN_2.save('CNN_2.h5')

#Carregando a RNC 2
from tensorflow import keras
CNN_2 = keras.models.load_model('CNN_2.h5')

```

```

#RNC3=====
#Criando a estrutura da RNC 3
CNN_3 = Sequential()
CNN_3.add(Conv2D(filters=6,
                 kernel_size=5,activation="relu",
                 input_shape=(256,256,3)))
CNN_3.add(MaxPool2D(pool_size=3,
                   padding="valid"))
CNN_3.add(Conv2D(filters=16,
                 kernel_size=3,activation="relu",
                 input_shape=(256,256,3)))
CNN_3.add(MaxPool2D(pool_size=3,
                   padding="valid"))
CNN_3.add(Flatten())
CNN_3.add(Dense(256,activation="relu"))
CNN_3.add(Dense(1,activation="relu"))
CNN_3.compile(loss='mean_squared_error',
              metrics=['mean_absolute_error'],
              optimizer = 'adam')
CNN_3.summary()

#Treinando a RNC 3
CNN_3.fit(X_train,y_train,batch_size = 5, epochs = 100)

#Salvando a RNC 3
CNN_3.save('CNN_3.h5')

#Carregando a RNC 3
from tensorflow import keras
CNN_3 = keras.models.load_model('CNN_3.h5')

```

APÊNDICE VI

```
#Avaliação estatística das redes neurais convolucionais=====
#Previsões para os dados de teste
previsoes_1 = CNN_1.predict(X_test)
previsoes_2 = CNN_2.predict(X_test)
previsoes_3 = CNN_3.predict(X_test)

# Avaliação estatística nos dados de teste
from sklearn.metrics import mean_absolute_error, mean_squared_error
mae_1 = mean_absolute_error(y_test, previsoes_1)
rmse_1 = np.sqrt(mean_squared_error(y_test, previsoes_1))
mae_2 = mean_absolute_error(y_test, previsoes_2)
rmse_2 = np.sqrt(mean_squared_error(y_test, previsoes_2))
mae_3 = mean_absolute_error(y_test, previsoes_3)
rmse_3 = np.sqrt(mean_squared_error(y_test, previsoes_3))

#Previsões para os dados de treinamento
previsoes_1 = CNN_1.predict(X_train)
previsoes_2 = CNN_2.predict(X_train)
previsoes_3 = CNN_3.predict(X_train)

#Avaliação estatística nos dados de treinamento
mae_1 = mean_absolute_error(y_train, previsoes_1)
rmse_1 = np.sqrt(mean_squared_error(y_train, previsoes_1))
mae_2 = mean_absolute_error(y_train, previsoes_2)
rmse_2 = np.sqrt(mean_squared_error(y_train, previsoes_2))
mae_3 = mean_absolute_error(y_train, previsoes_3)
rmse_3 = np.sqrt(mean_squared_error(y_train, previsoes_3))
```