



UNIVERSIDADE
ESTADUAL DE LONDRINA

EDSON LUIZ PILATI FILHO

MODELO DE APRENDIZADO DE MÁQUINA PARA
AVALIAÇÃO AUTOMÁTICA DO DESENVOLVIMENTO
MOTOR GROSSO

LONDRINA

2023

EDSON LUIZ PILATI FILHO

**MODELO DE APRENDIZADO DE MÁQUINA PARA
AVALIAÇÃO AUTOMÁTICA DO DESENVOLVIMENTO
MOTOR GROSSO**

Dissertação apresentada ao Programa de
Mestrado em Ciência da Computação da
Universidade Estadual de Londrina para ob-
tenção do título de Mestre em Ciência da
Computação.

Orientador: Prof. Dr. Jacques Duilio
Brancher

LONDRINA

2023

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema de Bibliotecas da UEL

P637m Pilati Filho, Edson Luiz.
Modelo de aprendizado de máquina para avaliação automática do desenvolvimento motor grosso / Edson Luiz Pilati Filho. - Londrina, 2023.
52 f. : il.

Orientador: Jacques Duílio Brancher.
Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual de Londrina, Centro de Ciências Exatas, Programa de Pós-Graduação em Ciência da Computação, 2023.
Inclui bibliografia.

1. Aprendizado de máquina - Tese. 2. Avaliação automática - Tese. 3. Desenvolvimento Motor - Tese. 4. TGMD-3 - Tese. I. Brancher, Jacques Duílio. II. Universidade Estadual de Londrina. Centro de Ciências Exatas. Programa de Pós-Graduação em Ciência da Computação. III. Título.

CDU 519

EDSON LUIZ PILATI FILHO

**MODELO DE APRENDIZADO DE MÁQUINA PARA
AVALIAÇÃO AUTOMÁTICA DO DESENVOLVIMENTO
MOTOR GROSSO**

Dissertação apresentada ao Programa de
Mestrado em Ciência da Computação da
Universidade Estadual de Londrina para ob-
tenção do título de Mestre em Ciência da
Computação.

BANCA EXAMINADORA

Orientador: Prof. Dr. Jacques Duilio
Brancher
Universidade Estadual de Londrina

Prof. Dr. André Luís Andrade Menolli
Universidade Estadual do Norte do Paraná
– UENP

Prof. Dr. Rodrigo Martins de Oliveira
Spinosa
Universidade Estadual de Londrina – UEL

Londrina, 29 de setembro de 2023.

*Este trabalho é dedicado à todos os
professores que, direta ou indiretamente,
contribuíram com a minha formação.*

AGRADECIMENTOS

Os Agradecimentos principais são direcionados ao Prof. Dr. Rodrigo Martins de Oliveira Spinosa que cedeu os vídeos para que esse trabalho fosse possível. Agradeço à minha família, que em nenhum momento deixou de me apoiar nas minhas decisões e sempre ofereceu todo o suporte para a minha formação. Agradeço também a todos os professores que contribuíram com a minha formação. Agradeço também ao meu orientador, Prof. Dr. Jacques Duilio Brancher, que me deu a oportunidade de escrever esta dissertação.

*“Diante do infinito, somos apenas um
pálido ponto azul.
(Carl Sagan)*

PILATI FILHO, E. L.. **Modelo de aprendizado de máquina para avaliação automática do desenvolvimento motor grosso**. 2023. 52f. Dissertação (Mestrado em Ciência da Computação) – Universidade Estadual de Londrina, Londrina, 2023.

RESUMO

As habilidades motoras grossas são atividades que envolvem os grandes músculos do corpo humano, como sentar, agachar, andar e correr. O *Test of Gross Motor Development* ou TGMD, é um teste muito utilizado por pesquisadores, pediatras, fisioterapeutas e educadores de diversos países para avaliar estas habilidades em crianças na faixa etária de 3 até 11 anos. Uma parte importante do teste é que o movimento executado pelas crianças precisa ser gravado e avaliado por dois ou mais profissionais. O processo de avaliação é trabalhoso e leva tempo, sendo sua automatização um dos principais pontos a serem desenvolvidos. Nos últimos anos, têm sido propostos métodos para automatizar a avaliação dos movimentos de acordo com os critérios do TGMD. A hipótese investigada neste trabalho é a de que é possível induzir um modelo de aprendizado de máquina capaz de identificar se o movimento da criança está correto. A proposta é avaliar automaticamente o primeiro critério da habilidade de salto do TGMD-3. Foi utilizada uma rede neural convolucional, *PoseNet*, para extrair o esqueleto das crianças, então foram modeladas características que representam o primeiro critério da habilidade de salto. *K-nearest neighbors*, *Random Forest* e *One-class SVM* foram testados para determinar sua acurácia na classificação do movimento, usando um *dataset* de 350 imagens de crianças brasileiras, entre 3 e 11 anos, realizando o movimento preparatório para o salto. Os resultados experimentais mostram uma acurácia média de 84% para o *K-nearest neighbors*, 84% de acurácia média para o *Random Forest* e 77% de acurácia média para o *One-class SVM*.

Palavras-chave: Avaliação automática. Aprendizado de máquina. Desenvolvimento Motor. TGMD-3.

PILATI FILHO, E. L.. **Machine learning model for automatic assessment of gross motor development**. 2023. 52p. Master's Thesis (Master in Science in Computer Science) – State University of Londrina, Londrina, 2023.

ABSTRACT

Gross motor skills are activities that involve the large muscles of the human body, such as sitting, crouching, walking and running. The Test of Gross Motor Development or TGMD, is a test widely used by researchers, pediatricians, physiotherapists and educators from different countries to assess these skills in children aged 3 up to 11 years old. An important part of the test is that the movement performed by the children needs to be recorded and evaluated by two or more professionals. The evaluation process is laborious and takes time, and its automation is one of the main points to be developed. In recent years, methods have been proposed to automate the assessment of movements according to the TGMD criteria. The hypothesis investigated in this work is that it is possible to induce a machine learning model capable of identifying whether the child's movement is correct. The proposal is to automatically assess the first criteria of the horizontal jump skill of the TGMD-3. PoseNet, a convolutional neural network, was used to extract the children's skeleton, then new features were modeled, which represents the first criterion of the jumping skill. K-nearest neighbors, Random Forest and One-class SVM were tested to determine their accuracy in classifying the movement, using a dataset of 350 images of Brazilian children, between 3 and 11 years old, performing the preparatory movement for the jump. The experimental results show an average accuracy of 84% for the K-nearest neighbors, 84% average accuracy for the Random Forest and 77% average accuracy for the One-class SVM.

Keywords: Automatic assessment. Machine Learning. Motor Development. TGMD-3

LISTA DE ILUSTRAÇÕES

Figura 1 – Vídeos de confiabilidade disponibilizados na página oficial do TGMD-3. (Fonte: [1])	22
Figura 2 – Critérios de avaliação da habilidade de salto adaptado do TGMD-3. (Fonte: Adaptado de [2])	23
Figura 3 – Pontos-chave do esqueleto humano por diferentes <i>datasets</i> : (a) COCO, e (b) MPII (Fonte: [3])	24
Figura 4 – Estimativa de pose: (a) Uma pessoa e (b) múltiplas pessoas. (Fonte: [4])	26
Figura 5 – Exemplo de estimativa top-down de poses.(Fonte: [3])	26
Figura 6 – Fluxo do modelo <i>PoseNet</i> . (Fonte: Adaptado de [5])	27
Figura 7 – Exemplo de árvore de decisão com os segmentos de decisão. (Fonte: Adaptado de [6])	29
Figura 8 – Exemplo de classificação com <i>K-nearest neighbors</i> . (Fonte: Adaptado de [7])	30
Figura 9 – Exemplo de margens de uma SVM. Os pontos sob as linhas tracejadas são os vetores de suporte. (Fonte: Adaptado de [8])	31
Figura 10 – Detecção de anomalia com <i>One-class SVM</i> com kernel RBF. (Fonte: [9])	31
Figura 11 – Exemplo de validação cruzada com <i>10-fold</i> . (Fonte: Adaptado de [10]) .	34
Figura 12 – Diagrama dos processos para obtenção do modelo. (Fonte: próprio autor)	35
Figura 13 – Exemplo de esqueleto do corpo de uma criança extraído com <i>PoseNet</i> . (Fonte: Próprio autor)	37
Figura 14 – Visualização dos dados com PCA de dois componentes. (Fonte: Próprio autor)	40
Figura 15 – Matriz de confusão do melhor modelo KNN nos dados de <i>holdout</i> (eixo vertical são os dados reais). (Fonte: próprio autor)	42
Figura 16 – Matriz de confusão do melhor modelo <i>Random Forest</i> nos dados de <i>holdout</i> (eixo vertical são os dados reais). (Fonte: próprio autor)	43

LISTA DE TABELAS

Tabela 1 – Diferenças no reconhecimento de pontos entre COCO e MPII.	25
Tabela 2 – Matriz de confusão	32
Tabela 3 – Performance dos modelos do KNN de validação cruzada	41
Tabela 4 – Performance dos modelos do <i>Random Forest</i> de validação cruzada . .	42

LISTA DE ABREVIATURAS E SIGLAS

CNN	Convolution Neural Network
COCO	Common Objects in Context
FMS	Fundamental Motor Skill
GMA	Gross Motor Assessment
GMS	Gross Motor Skills
HMF	Habilidade Motora Fundamental
IMU	Inertial Measurement Unit
KTK	Körperkoordination Test für Kinder
MABC	Movement Assessment Battery for Children
PCA	Principal Component Analysis
RBF	Radial Basis Function
SVM	Support Vector Machine
TGMD	<i>Test of Gross Motor Development</i>

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Problemas de pesquisa	15
1.2	Objetivos	15
1.2.1	Objetivo Geral	15
1.2.2	Objetivos Específicos	15
1.3	Estrutura da dissertação	16
2	TRABALHOS RELACIONADOS	17
3	FUNDAMENTAÇÃO TEÓRICA	20
3.1	Desenvolvimento Motor na infância	20
3.2	Aspectos teóricos e práticos do TGMD	21
3.3	Modelo de estimativa de poses	24
3.3.1	Redes neurais convolucionais	27
3.3.2	<i>PoseNet</i>	27
3.4	Algoritmos de classificação	28
3.4.1	Árvore de decisão	28
3.4.2	<i>K-nearest neighbors</i>	30
3.4.3	Máquina de Vetor de Suporte	30
3.5	Métricas de avaliação	32
3.5.1	Matriz de confusão	32
3.5.2	Acurácia	32
3.5.3	Precisão	33
3.5.4	<i>Recall-Score</i> ou sensibilidade	33
3.5.5	F-Score	33
3.6	Validação Cruzada	33
3.6.1	Validação cruzada com <i>K-fold</i>	33
4	MATERIAIS E MÉTODOS	35
4.1	<i>Dataset</i>	35
4.2	Extração do esqueleto com <i>PoseNet</i>	36
4.3	Seleção de características	37
4.4	Visualização dos dados	39
4.5	Separação do <i>dataset</i>	39
5	RESULTADOS E DISCUSSÕES	41

6	CONCLUSÃO	45
	REFERÊNCIAS	47
	Trabalhos Publicados pelo Autor	52

1 INTRODUÇÃO

O Desenvolvimento motor é frequentemente dividido em habilidades motoras grossas e habilidades motoras finas. As habilidades motoras grossas são atividades que envolvem os grandes músculos do corpo humano, como sentar, agachar, andar e correr. As habilidades motoras finas são aquelas atividades que envolvem músculos menores, como manipulação de objetos, agarrar e escrever. [11]

Formalmente, o Desenvolvimento Motor pode ser definido como as mudanças no comportamento motor que ocorrem ao longo da vida [12][13]. A compreensão desse fenômeno, principalmente na fase infantil, depende da observação e avaliação das alterações dessas classes de movimentos, implicando em identificar, descrever e mensurar comportamentos em vários momentos [14].

As habilidades básicas são necessárias para manter uma rotina diária em casa e na escola, como também serve a propósitos lúdicos, pois já nos primeiros anos de vida, a cultura requer das crianças um domínio de várias habilidades, principalmente no início do processo de escolarização. [15]

Dentre os protocolos para avaliação do desenvolvimento motor, destaca-se o TGMD, por ser um instrumento destinado a avaliar as habilidades motoras fundamentais de crianças com idade entre 3 e 11 anos [2] possuindo procedimentos padronizados e validados para a população brasileira. O teste é de fácil aplicação, sendo utilizado a mais de 30 anos por pesquisadores, pediatras, fisioterapeutas e educadores de diversos países, tanto para avaliar níveis de desenvolvimento motor, como para auxiliar no planejamento de programas interventivos em escolas [16] [2].

No modelo de aplicação do teste TGMD-3, a tarefa de avaliação pode se tornar complexa e suscetível ao erro humano. Todas as crianças precisam ser gravadas. O avaliador precisa ser muito minucioso na análise dos pontos-chave do vídeo que correspondem aos critérios de desempenho. A quantidade de análises necessárias pode ser extremamente grande, fazendo com que a avaliação se torne morosa, considerando que cada participante do teste gera 26 vídeos que devem ser pontuados individualmente.

Nesse sentido, a automatização do processo se torna um caminho muito natural e pertinente para acelerar o processo de correção e potencializar a confiabilidade do resultado. Alguns autores já propuseram modelos de avaliação automática das habilidades do TGMD. Em [17] foi proposta uma avaliação através de sensores IMUs (*inertial measurement unit*) acoplados nas costas, nos tornozelos e pulsos. Em [18] os autores apresentam uma abordagem mais visual, mostrando ao avaliador os pontos do esqueleto onde provavelmente a criança executou o movimento de forma errada.

Esta dissertação apresenta uma nova abordagem de avaliação automática do primeiro critério da habilidade de salto do TGMD-3 utilizando *deep learning*. Essa abordagem inclui a utilização do *PoseNet*, uma rede neural convolucional (CNN), para extração do esqueleto da criança a partir de uma imagem. Redes neurais convolucionais são adequadas para o problema, pois tratam de processamento de imagens em nível de reconhecimento de poses. As CNNs são particularmente conhecidas por seu bom desempenho nesses tipos de problemas que envolvem a visão computacional.

Também foi proposta uma modelagem de características para representar o primeiro critério da habilidade de salto do TGMD-3. Essa modelagem tem como objetivo selecionar características para separar bem o problema, e possibilitar a criação de um modelo de aprendizado de máquina para atribuir a nota do critério automaticamente.

A abordagem apresentada neste trabalho se destaca por não ser necessário nenhum equipamento acoplado à criança durante a execução da habilidade. Foi necessário apenas extrair a imagem do momento que representa o primeiro critério. Além disso, a saída do modelo gerado por essa abordagem é diretamente a nota do critério.

1.1 Problemas de pesquisa

É possível um modelo, baseado em aprendizado de máquina, identificar se o movimento da criança está correto para o primeiro critério da habilidade de salto do TGMD-3?

1.2 Objetivos

1.2.1 Objetivo Geral

Induzir um modelo de aprendizado de máquina que seja capaz de identificar se o movimento da criança está correto para o primeiro critério da habilidade de salto do TGMD-3.

1.2.2 Objetivos Específicos

O primeiro objetivo específico é extrair o esqueleto das crianças com o *PoseNet* e montar um *dataset* com os pontos de articulação do corpo.

O segundo objetivo específico é modelar novas características para representar bem o primeiro critério do movimento de salto como é descrito no TGMD-3.

O terceiro objetivo específico é gerar modelos de aprendizado de máquina treinados a partir das características selecionadas.

1.3 Estrutura da dissertação

Este capítulo apresentou uma introdução ao tema deste trabalho. Os capítulos seguintes da dissertação são divididos da seguinte maneira. No capítulo 2 são apresentados os principais trabalhos relacionados para automatização da avaliação do TGMD. No capítulo 3 são apresentadas as fundamentações teóricas do trabalho. No capítulo 4 são apresentadas as contribuições do trabalho com seus métodos e materiais utilizados. No capítulo 5 são apresentados e discutidos os resultados deste trabalho e no capítulo 6 são apresentadas as conclusões.

2 TRABALHOS RELACIONADOS

Neste capítulo são discutidos alguns trabalhos que se relacionam com este no sentido de automatizar a avaliação do desenvolvimento motor seguindo os critérios do TGMD. É uma evolução natural dos testes de avaliação do desenvolvimento motor tendo em vista os problemas enfrentados pelos profissionais na hora de atribuir as notas do teste.

Em [19] foi desenvolvido um sistema de suporte para análise do desenvolvimento motor de crianças japonesas implementando o TGMD-2. Neste primeiro trabalho os autores desenvolveram um sistema chamado *Gross Motor Assessment* (GMA) para auxiliar os profissionais na avaliação dos vídeos de crianças executando as habilidades do TGMD-2.

O objetivo deste trabalho foi lançar as bases para um aplicativo que, posteriormente, foi melhorado no sentido de fornecer um diagnóstico automático, mostrando os pontos onde possivelmente a criança executou o movimento de forma errada.

Para o trabalho, foram gravadas 24 crianças japonesas em dois ângulos, ou seja, com duas câmeras, gerando assim dois vídeos. Cada criança executou as 12 habilidades do TGMD-2 por 2 ou 3 vezes. Todas as crianças foram gravadas no mesmo vídeo, então como resultado obteve-se dois vídeos longos, representando duas perspectivas diferentes, contendo todas as habilidades executadas por todas as crianças. Além disso, foi acoplado um medidor de aceleração no braço dominante da criança. Esse medidor gera um gráfico de vibrações e impactos durante a execução da habilidade, desta forma indicando os picos de intensidade na execução dos movimentos.

Com os vídeos disponíveis, foi necessário que um profissional capacitado sincronizasse o gráfico de aceleração e os dois vídeos das duas perspectivas, e fizesse o *upload* dos vídeos na plataforma GMA que desenvolveram. Depois dos vídeos disponíveis na aplicação, é preciso que se identifique os pontos do vídeo onde cada criança executou cada habilidade. Dessa forma, marcando o tempo do vídeo em que determinada criança executou determinada habilidade.

Depois das marcações, é possível facilmente navegar na aplicação, procurando as crianças desejadas, e também a habilidade desejada, dessa forma facilitando a avaliação de cada criança. Além disso, é possível visualizar o gráfico de aceleração da criança durante a execução da habilidade. O avaliador pode então dar uma nota segundo os critérios do TGMD-2. Todos estes dados ficam salvos no aplicativo para posterior consulta ou impressão de relatórios.

O foco principal deste primeiro trabalho dos autores em [19], foi justamente diminuir o tempo que o profissional avaliador leva para dar notas segundo o TGMD-2. O objetivo foi atingido neste trabalho, resultando em uma avaliação de 3 à 5 minutos por

criança. Porém, pode-se indicar que o processo de *upload* dos vídeos e marcação dos pontos são processos extremamente trabalhosos e requerem profissionais capacitados para tal tarefa.

Os mesmo autores em [20] utilizaram o *OpenPose*, um modelo de reconhecimento de poses, para extrair o esqueleto das crianças e implementar uma classificação automática das habilidades dentro do aplicativo criado no trabalho anterior. Neste trabalho já foi utilizada a versão 3 do TGMD, sendo assim, foram classificadas as 13 habilidades contidas no teste. A classificação referida aqui não é da nota, mas sim a detecção de qual habilidade se refere o vídeo da criança, de forma que os profissionais não precisem identificar manualmente se habilidade é de salto, arremesso, etc. A acurácia da classificação da habilidade foi de 82.3% utilizando um *stratified 8-fold cross validation*. A etapa de avaliação automática dos critérios das habilidades foi citada como um trabalho futuro.

No trabalho seguinte dos mesmos autores, também não é tratada a automatização dos critérios do TGMD-3, sendo referida como trabalho futuro. Neste trabalho o modelo de classificação das habilidades é melhorado com uma estratégia de aumentar o *dataset*. A acurácia foi melhorada para 99.5% na classificação de habilidades com *cross validation* [21].

É importante ressaltar que nos trabalhos [19], [20] e [21] a automatização dos critérios do TGMD-3 é citada sempre como um diagnóstico do movimento, ou seja, não necessariamente é considerada atribuição da nota de forma automática, e sim a realização de um diagnóstico indicando onde, provavelmente, a criança executou um critério de forma incorreta, o que de certa forma, se trata de uma automação.

Em [18], os mesmos autores propõe uma forma de automatização da avaliação dos critérios das habilidades do TGMD-3. No trabalho é dado sequência ao melhoramento do aplicativo desenvolvido pelos autores, mas o foco do trabalho é apresentar uma proposta de diagnóstico das habilidades do TGMD-3. A estratégia foi utilizar a técnica de detecção de anomalia, uma forma não supervisionada, em que os autores treinaram um modelo com movimentos corretos, e tudo que diverge desses movimentos é considerado anomalia.

Dessa forma, o esqueleto da criança é apresentado com cores, de forma que as partes do corpo, onde possivelmente a criança executou o movimento de forma errada, são apresentadas em vermelho. Essa alternativa é como um mapa de calor, não apresenta diretamente a nota de forma automática, apenas fornece ao avaliador os possíveis pontos de falha no movimento executado pela criança. Os resultados apontam que foi possível enfatizar visualmente os pontos que correspondem aos critérios do TGMD-3.

É possível verificar que os trabalhos citados acima, tratam mais das formas de facilitar o processo de avaliação para o profissional avaliador. Apesar de fornecer um diagnóstico do movimento em forma de mapa de calor, a atribuição automática da nota

não é feita. Dessa forma, nota-se a importância dessa dissertação no sentido de avançar na automatização da avaliação dos critérios do TGMD-3, além disso, a proposta dessa dissertação não requer equipamentos que precisam ser acoplados à criança durante a execução das habilidades, o que torna o modelo mais simples de ser adotado.

Em [17] para avaliar a competência de movimento das crianças de acordo com TGMD-2, foi feito um estudo com 45 crianças italianas de 6 a 10 anos de idade. Foram utilizados cinco sensores IMUs (*inertial measurement unit*) que foram montados nas costas, nos tornozelos e nos pulsos para capturar os movimentos.

Foram capturados os movimentos em vídeo e foram submetidos a 3 profissionais avaliadores para fornecerem uma nota segundo o TGMD-2. Em paralelo, foi desenvolvido um algoritmo para análise automática dos vídeos com os dados capturados pelos sensores, os resultados apontam para uma redução no tempo de 15 minutos para 2 minutos de avaliação, mas carece de melhorias na qualidade, mostrando uma concordância de mais de 87% quando comparada com a técnica de avaliação tradicional [17]. No entanto, essa técnica é limitada em relação a proposta desta dissertação, pois é obrigatória a utilização dos sensores e equipamentos para realizar a avaliação.

3 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão abordadas as fundamentações teóricas para o TGMD e as tecnologias utilizadas neste trabalho.

3.1 Desenvolvimento Motor na infância

É possível caracterizar o desenvolvimento motor na infância como a aquisição de uma série de habilidades com amplo espectro que possibilita a criança um domínio do seu corpo em diferentes posturas (estáticas e dinâmicas), variadas formas de locomover-se (andar, correr, saltar, etc.) e manipulação de objetos e instrumentos variados (receber uma bola, arremessar uma pedra, chutar, escrever, etc.). [15]

O desenvolvimento motor é um fenômeno físico que ocorre por meio das atividades da criança em ambientes físicos específicos. Está entrelaçado com as interações sociais e é afetado pelo contexto sociocultural mais amplo. [22] A habilidade motora pode ser descrita como o aprendizado de sequências de movimentos que combinados produzem uma constante e eficiente ação para domínio de uma tarefa. [23]

Instrumentos para avaliação do desenvolvimento motor na infância são amplamente utilizados na pesquisa científica e na prática clínica. Os instrumentos permitem que profissionais possam identificar precocemente e entender os mecanismos e as alterações do desenvolvimento psicomotor infantil, além de ser uma ferramenta auxiliar de triagem diagnóstica. [24]

O diagnóstico do desenvolvimento motor permite que os profissionais possam identificar os causadores de limitação no movimento, auxiliando na tomada de decisão sobre quais habilidades ou critérios devem ser reforçados nos programas; o tempo de prática destinado a cada habilidade e os objetivos de desempenho de cada criança. [25]

Nesse sentido, os instrumentos de avaliação tornam-se ferramentas de grande utilidade para acompanhar o desenvolvimento geral do indivíduo. Na educação infantil o uso de instrumentos de avaliação das habilidades motoras fundamentais (HMFs) é imprescindível, pois, permite aos professores e técnicos: determinar o estado de desenvolvimento das HMFs da criança, monitorar o progresso da criança diante de programas de habilidades motoras; examinar a efetividade dos programas e garantir a adequação entre as atividades executadas e o desenvolvimento da criança [26].

Diferentes protocolos foram desenvolvidos para avaliação motora infantil, como o *Körperkoordination Test für Kinder* (KTK) [27], a escala de desenvolvimento motor [28], o *Movement Assessment Battery for Children* (MABC) [29], o *Test of Gross Motor*

Development (TGMD) [30] [2], entre outros.

3.2 Aspectos teóricos e práticos do TGMD

O TGMD (*Test of Gross Motor Development*) teve origem em dissertação de Ulrich em 1981 e foi publicado em 1985. Existia uma necessidade grande para desenvolvimento de instrumentos de avaliação das habilidades motoras que combinassem com o currículo físico da educação de base.

Um dos conteúdos mais comuns antes de sua criação era o HMF (habilidades motoras fundamentais), que prevê habilidades desenvolvidas no começo da infância e continuam seu desenvolvimento por todo o processo de crescimento [2]. Essas habilidades são descritas como sendo adquiridas por crianças de diferentes culturas ao redor do mundo e se tornam a base para habilidades especializadas utilizadas para prática de esportes ou jogos. [31]

Ulrich desenvolveu uma lista de habilidades motoras fundamentais nas fontes bibliográficas da época, incluindo uma listagem de critérios qualitativos usados para cada habilidade. A lista de habilidades foi submetida a vários profissionais da área e professores de educação física da educação de base de crianças para avaliar cada habilidade e seus critérios, em um esforço para estabelecer uma validação apropriada. As habilidades e critérios obtiveram 90% de concordância entre os profissionais e professores consultados.[2]

Devido às recomendações de que testes padronizados que utilizam normas como referências devem ser atualizados a cada 15 anos, Ulrich publica a segunda versão do teste TGMD em 2000 [30]. Um grande desafio foi descrever os critérios de desempenho de forma que os profissionais aplicadores do teste entendam e consigam determinar a validade do movimento executado pela criança. Portanto, ao longo de suas 3 versões as maiores alterações foram para clarificar as descrições dos critérios de desempenho. [2]

Na terceira versão do *Test of Gross Motor Development* (TGMD-3), foram renomeados grupos de atividades para melhor compreensão por parte dos profissionais aplicadores do teste. Algumas adaptações também foram feitas para aplicação do teste em crianças com autismo.

Em sua terceira versão são propostos a avaliação de 13 habilidades divididas em duas classes de movimentos, sendo elas: locomotoras (correr, galopar, pular com um pé só, saltar horizontalmente, correr lateralmente e saltitar) e manipulativas (quicar, rebater com duas mãos, rebater com uma mão, arremessar por cima, arremessar por baixo, receber e chutar) [2].

A habilidade de rolar a bola por baixo foi removida e adicionada a habilidade de arremesso por baixo devido à dificuldade demonstrada por muitas crianças na execução

completa de todos os critérios. Além disso, a habilidade de arremesso por baixo é mais utilizada em jogos e esportes.

Foi adicionada a rebatida com apenas uma mão. Devido à grande aderência ao TGMD-2 globalmente, foi verificado que crianças, especialmente em países asiáticos, nem sempre desenvolvem a rebatida com duas mãos, enquanto a rebatida com uma mão é comum em esportes com raquete em todo o mundo.

No geral, devido a pesquisas em saúde pública, verificou-se que as competências relacionadas a habilidades com bola tem relação com o nível de atividade física no futuro da criança, optou-se por aumentar o número de habilidades com bola no TGMD-3. Como resultado, o TGMD-3 apresenta seis habilidades de locomoção e sete habilidades com bola. [2]

Devido à necessidade de uma boa demonstração das habilidades do teste, os próprios criadores desenvolveram um *website*, como mostra a figura 1, onde disponibilizam vídeos de como executar todas as habilidades contidas no TGMD-3 [1].

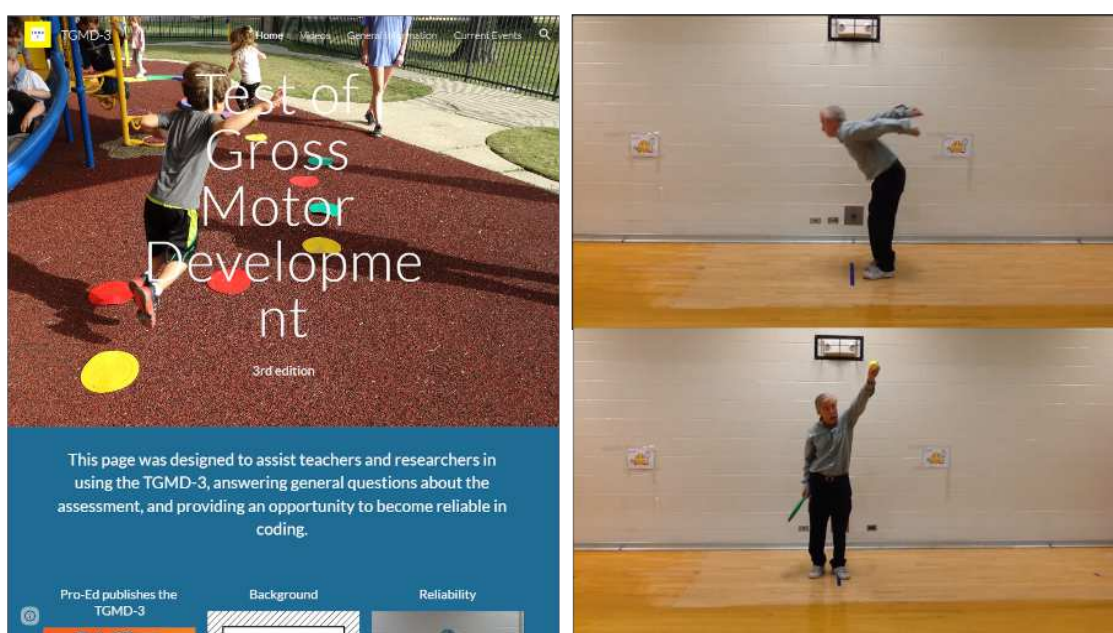


Figura 1 – Vídeos de confiabilidade disponibilizados na página oficial do TGMD-3. (Fonte: [1])

Para aplicação do teste os avaliadores devem se atentar aos procedimentos propostos pelo manual técnico, que inclui a demonstração eficiente juntamente com a instrução verbal da habilidade que será realizada, uma tentativa de familiarização da criança avaliada e duas tentativas válidas que serão filmadas para posterior análise e pontuação de especialistas.

Cada movimento possui entre 3 e 5 critérios de desempenho que devem ser executados para contabilizar a pontuação de desempenho da criança. Cada critério correto recebe a nota 1, e cada critério incorreto recebe nota 0, gerando a nota parcial da habilidade motora [2].

A filmagem do participante é imprescindível para a etapa de análise, em que dois ou mais especialistas assistem cuidadosamente à gravação de todos os vídeos das crianças, observando se durante a realização de cada habilidade motora os critérios de desempenho correspondente foram executados [2].

Uma das principais habilidades avaliadas pelo TGMD-3 é a habilidade de saltar, pois é fácil de ser executada por todas as crianças, já que não necessita de objetos para sua realização. Para este trabalho foram utilizadas as filmagens da habilidade de salto realizadas no trabalho de Spinosa [32].

A habilidade de salto é composta por 4 critérios. O primeiro é o movimento preparatório que inclui flexão dos dois joelhos e os braços estendidos atrás do corpo. No segundo critério os braços são estendidos com força para frente e para cima, atingindo uma extensão máxima acima da cabeça. No terceiro critério, a criança deve levantar voo e aterrissar (tocar o solo) com os dois pés simultaneamente. No quarto critério os braços são trazidos para baixo durante a aterrissagem.

Neste trabalho o objetivo é apenas desenvolver os experimentos em cima do primeiro critério da habilidade de salto do TGMD-3. Na figura 2 é possível verificar os critérios da habilidade de salto.

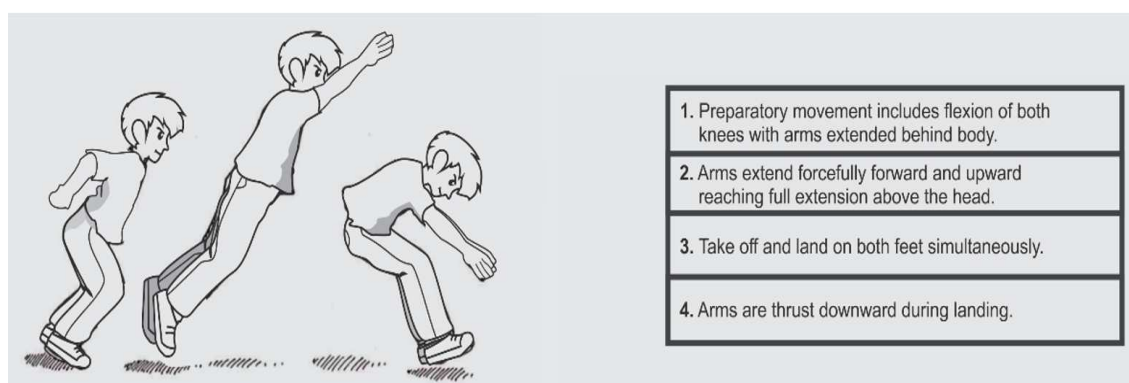


Figura 2 – Critérios de avaliação da habilidade de salto adaptado do TGMD-3. (Fonte: Adaptado de [2])

O TGMD é muito aplicado para crianças, logo estudos para expandir sua aplicação também foram feitos. Em [33] foi realizado um estudo com 844 participantes para validar propostas de estender a aplicação do teste TGMD-2 em adolescentes, reduzindo o número de habilidades para sete, pois algumas habilidades tiveram baixa contribuição no resultado, o resultado apresentou bom ajuste.

O TGMD-3 é um instrumento validado para a população brasileira em [34] [35] onde foi feita uma tradução do teste para o português e envolveu 597 crianças entre 3 e 10 anos de idade. Os resultados apontam para a validade do teste e sua confiabilidade na avaliação do desenvolvimento motor de crianças brasileiras.

3.3 Modelo de estimativa de poses

A estimativa de poses trata de modelos capazes de prever os pontos-chave do corpo humano em uma imagem ou vídeo. Assim formando um esqueleto 2D ou 3D do corpo humano. É útil em muitas áreas da visão computacional como reconhecimento de ações humanas, rastreamento humano, interação humano-computador, jogos, linguagem de sinais e vigilância por vídeo [3].

O processo de estimar uma pose humana pode ser dividido em duas partes. A primeira é localizar os pontos de articulação ou pontos-chave do corpo humano e o segundo é agrupar esses pontos em poses válidas para um humano [3]. No primeiro passo é possível montar um esqueleto do corpo humano com os pontos identificados como mostra a figura 3.



Figura 3 – Pontos-chave do esqueleto humano por diferentes *datasets*: (a) COCO, e (b) MPII (Fonte: [3])

Os modelos de estimativa de poses podem ser treinados com diferentes *datasets*, isso possibilita que o resultado fornecido pelo modelo seja de diferentes pontos dependendo do *dataset*. Dois *datasets* muito conhecidos para reconhecimento de poses são o *Microsoft COCO (Common Objects in Context)* [36] e o MPII [37]. Na figura 3 pode-se verificar as diferenças de pontos reconhecidos entre os modelos treinados pelos dois *datasets*.

O modelo treinado com o *dataset* COCO pode fornecer 17 pontos-chave do corpo humano, enquanto que o modelo treinado com o *dataset* MPII pode fornecer 14 pontos-chave do corpo humano. Uma comparação entre os pontos reconhecidos por cada um dos *datasets* é fornecida na Tabela 1.

Tabela 1 – Diferenças no reconhecimento de pontos entre COCO e MPII.

COCO	MPII
Nariz - 0	Cabeça - 0
Pescoço - 1	Pescoço - 1
Ombro direito - 2	Ombro direito - 2
Cotovelo direito - 3	Cotovelo direito - 3
Pulso direito - 4	Pulso direito - 4
Ombro esquerdo - 5	Ombro esquerdo - 5
Cotovelo esquerdo - 6	Cotovelo esquerdo - 6
Pulso esquerdo - 7	Pulso esquerdo - 7
Quadril direito - 8	Quadril direito - 8
Joelho direito - 9	Joelho direito - 9
Tornozelo direito - 10	Tornozelo direito - 10
Quadril esquerdo - 11	Quadril esquerdo - 11
Joelho esquerdo - 12	Joelho esquerdo - 12
Tornozelo esquerdo - 13	Tornozelo esquerdo - 13
Olho direito - 14	Peito - 14
Olho esquerdo - 15	
Orelha direita - 16	
Orelha esquerda - 17	

A estimativa de poses pode ser feita tanto para uma pessoa quanto para múltiplos indivíduos presentes em uma imagem. A identificação da pose de uma pessoa é relativamente mais fácil quando comparada a detecção de múltiplas pessoas [38]. Na figura 4 é possível identificar um exemplo de detecção de apenas uma pessoa e um exemplo de estimativa de poses com múltiplas pessoas.

Há dois tipos de abordagens para estimativa de poses. A abordagem *top-down* identifica aproximadamente onde existe um humano na imagem e define uma caixa delimitadora ao redor, é então aplicado um reconhecimento de pose na região delimitada. Na figura 5 é apresentado um exemplo de reconhecimento *top-down*. A abordagem *bottom-up* localiza os pontos-chave que conseguir encontrar na imagem e em seguida realiza o agrupamento dos pontos em uma pose completa.

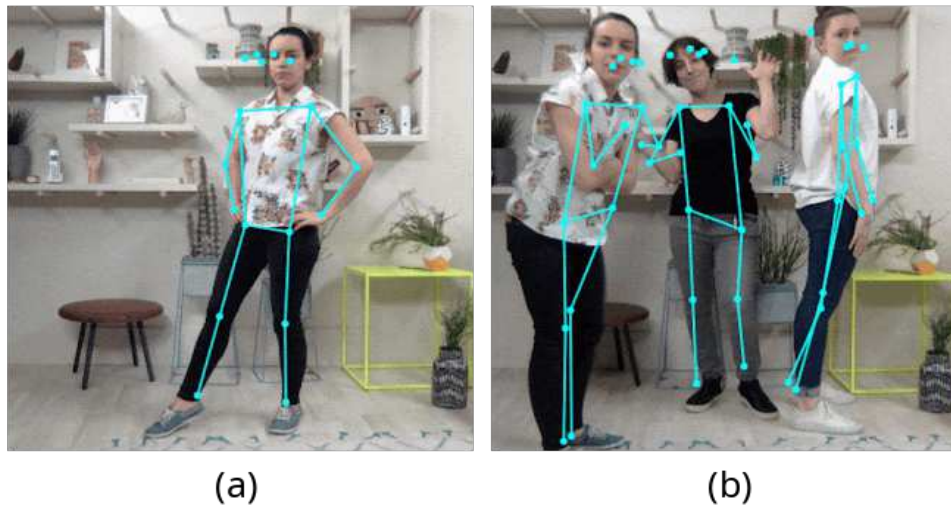


Figura 4 – Estimativa de pose: (a) Uma pessoa e (b) múltiplas pessoas. (Fonte: [4])

O *PoseNet*, modelo de estimativa de poses escolhido para este trabalho, utiliza a abordagem *bottom-up* com uma rede neural convolucional (CNN) onde o custo é independente do número de pessoas presentes na mesma imagem. A figura 4 é um exemplo de reconhecimento *bottom-up*. [39]

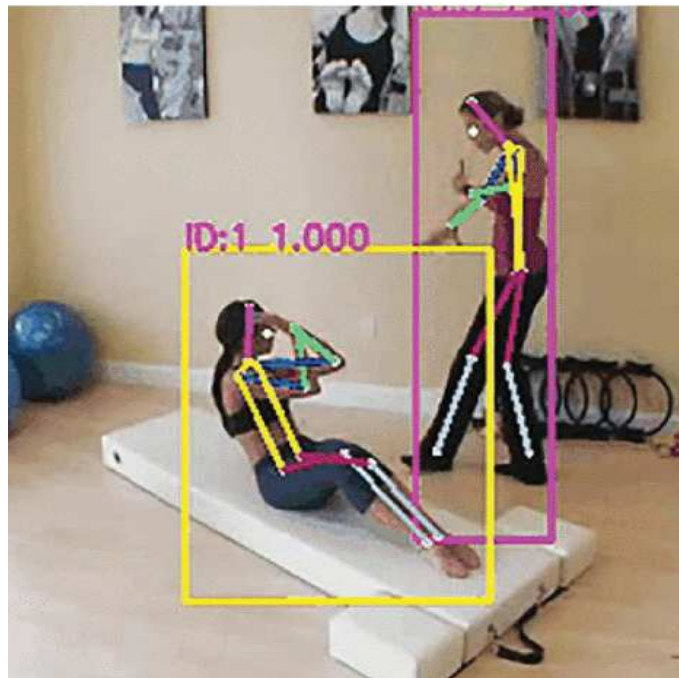


Figura 5 – Exemplo de estimativa top-down de poses.(Fonte: [3])

3.3.1 Redes neurais convolucionais

Muitas vezes nas técnicas de *deep learning*, principalmente no processamento de imagens, é muito custoso varrer todos os píxeis das imagens. Assim, procurando por um método mais eficiente, emerge que ao invés de uma varredura completa, é uma boa ideia olhar para regiões locais de uma imagem e não mais para uma imagem completa. Com isso, houve uma redução grande no custo computacional de redes neurais [40].

Redes neurais convolucionais (CNN) são redes neurais bem avançadas e com alto potencial para tratar de dados com alta complexidade e pré-processamento. Utiliza como referência o arranjo dos neurônios do córtex visual de um cérebro animal e é um dos mais eficientes e poderosos modelos para processamentos de imagens. As CNNs são muito utilizadas para processamento de imagens que envolvem análise, reconhecimento ou segmentação. [41]

3.3.2 PoseNet

Nesse contexto, um modelo de aprendizado de máquina de código aberto chamado *PoseNet* foi desenvolvido pelo Google Creative Lab [4]. O modelo é capaz de reconhecer poses em tempo real através de implementações que utilizam a câmera, ou através da alimentação com vídeos gravados e também imagens. O modelo é capaz de reconhecer 17 pontos-chave do corpo humano e múltiplas poses na mesma imagem [39]. O modelo foi treinado com o *dataset* COCO, muito conhecido para indução de modelos de reconhecimento de poses [36]. Na figura 6 é representado o fluxo do modelo *PoseNet*.

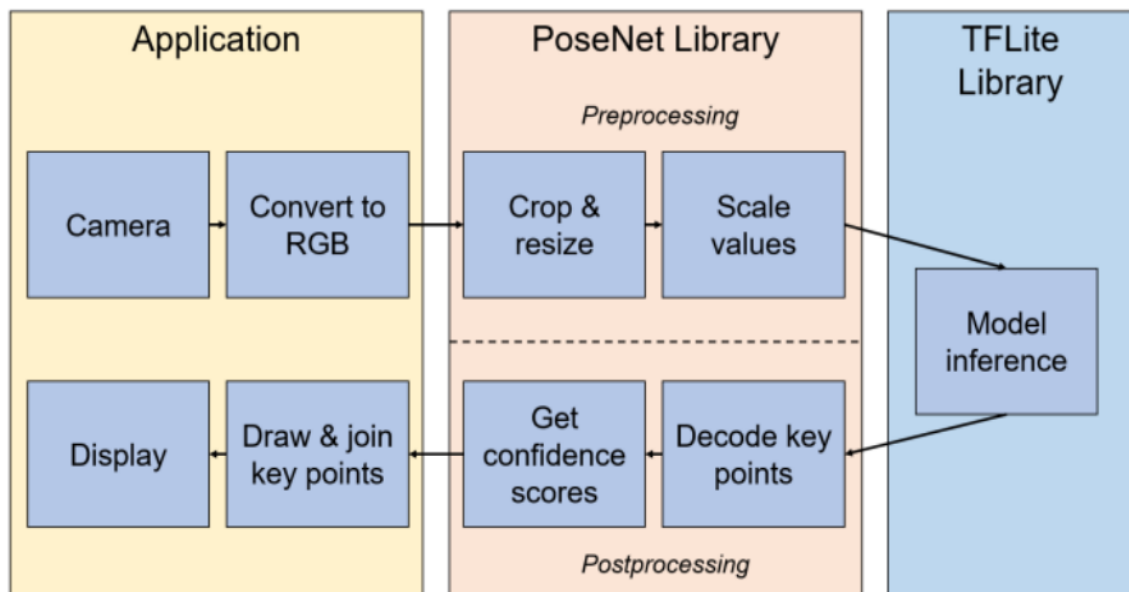


Figura 6 – Fluxo do modelo *PoseNet*. (Fonte: Adaptado de [5])

Em [42], foi feito um estudo sobre utilização do *PoseNet* como forma de extrair os pontos de articulação do corpo humano com o intuito de identificar atividades humanas. Para este trabalho foi desenvolvido um sistema. O estudo aponta resultados promissores no uso da técnica e avanços significantes com relação ao estado da arte.

Em [5] foi realizado um estudo para comparar os modelos *PoseNet*, *MoveNet* e *OpenPose* em dispositivos *mobile*. As imagens para testes foram divididas em três grupos, o primeiro com imagens contendo uma única pessoa, o segundo grupo com imagens contendo múltiplas pessoas e o terceiro grupo com imagens sem nenhuma pessoa. O terceiro grupo contendo imagens de animais e objetos foi adotado para identificar se algum dos modelos estima poses, mesmo quando não há humanos para identificar na imagem.

Os resultados em [5] mostram que o modelo que foi mais rápido em estimar as poses foi o *MoveNet*, enquanto o *OpenPose* foi o mais lento. O modelo que obteve a maior acurácia foi o *PoseNet*.

3.4 Algoritmos de classificação

Nesta seção são apresentados os algoritmos escolhidos para realizar os experimentos e gerar modelos de aprendizado de máquina para obtenção da nota no primeiro critério da habilidade de salto do TGMD-3.

3.4.1 Árvore de decisão

Uma árvore de decisão pode ser entendida como uma árvore binária que divide o *dataset* recursivamente usando nós de decisão até que seja encontrado um nó final, também chamado de folha, que representa a classe a ser predita. Uma árvore de decisão encontra a melhor divisão maximizando o ganho de informação [43].

Existem dois tipos de nós em uma árvore de decisão, os nós de decisão e os nós folha. Os nós de decisão contém uma condição que divide os dados, os dados são divididos sucessivamente até encontrar um nó folha. O nó inicial da árvore é chamado de nó raiz [44].

A medida que os dados vão sendo classificados, os nós vão recebendo informações, quanto mais um dado de uma classe específica chega a um nó, mais esse nó ganha informações da classe. Para saber qual separação é a melhor, a árvore de decisão maximiza o ganho de informação. Nem sempre um nó conterá informações de uma só classe, assim, o ganho de informação pode ser compreendido como um nó onde é possível distinguir claramente uma classe dominante. Basicamente, quanto mais uma classe é dominante no nó, mais ganho de informação a divisão possui [44].

A figura 7 exemplifica o processo seguido por uma árvore de decisão. Cada decisão segmenta uma parte do *dataset* em uma região específica.

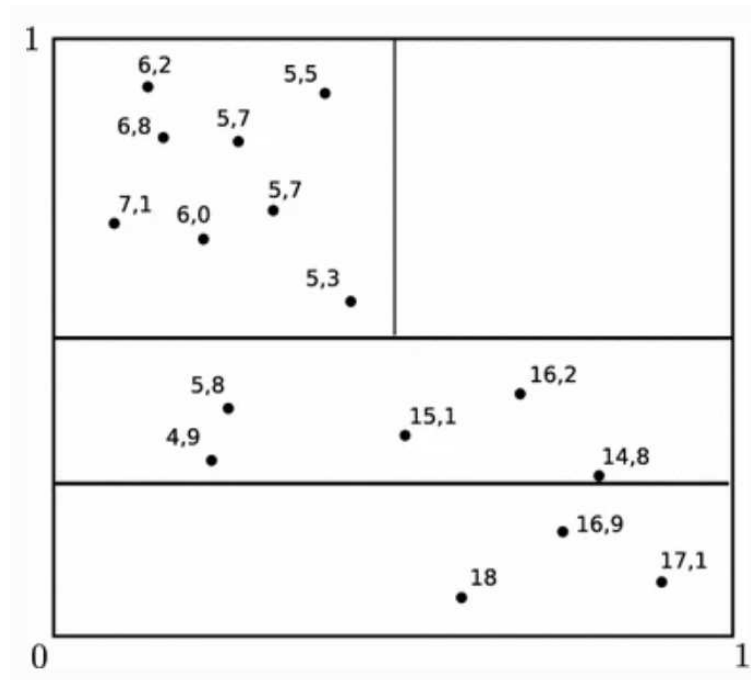


Figura 7 – Exemplo de árvore de decisão com os segmentos de decisão. (Fonte: Adaptado de [6])

As árvores de decisão são muito sensíveis aos dados de treino, o que pode resultar em uma alta variabilidade nos resultados, então o modelo gerado por uma árvore de decisão pode falhar em uma generalização maior.

O algoritmo *Random Forest* é formado por múltiplas árvores de decisão randômicas. Esse algoritmo é bem menos suscetível a variações, pois é menos sensível às mudanças nos dados de treino. O algoritmo consiste em gerar múltiplas árvores a partir de *sub-datasets* randômicos gerados a partir do *dataset* de treino, cada novo *sub-dataset* gera uma nova árvore. Para prever a classe de um novo dado, a *Random Forest* submete esse dado à todas as árvores geradas, assim, o novo dado é predito como sendo da classe majoritária entre as classes preditas pelas árvores. [45]

Nos processos de uma *Random Forest*, defini-se como *bagging*, em árvores de decisão, a criação de *sub-datasets* amostrais a partir dos dados de treino, o que é chamado de amostras *bootstrap*, então é gerada uma árvore de decisão para cada um dos *sub-datasets*, por fim, é realizada a agregação das árvores de decisão numa saída final que é a predição.

Foi utilizada a implementação do *sklearn* do *Random Forest* [46][47] e desenvolvido em linguagem *Python* [48] utilizando o *Jupyter Notebook* [49].

3.4.2 *K*-nearest neighbors

O algoritmo *K*-nearest neighbors (KNN) é um dos mais simples e conhecidos algoritmos de classificação. Ele se baseia na vizinhança do ponto para definir sua classe. Dada a entrada de um novo dado de teste, são verificados os vizinhos do ponto. Se estiver mais perto de dados com classe 1, esse ponto é considerado da mesma classe, e o mesmo vale para a classe 0 [50].

Foi utilizada a implementação do *sklearn* do KNN [46][51] e desenvolvido em linguagem *Python* [48] utilizando o *Jupyter Notebook* [49]. Na figura 8 é possível verificar um exemplo de utilização do KNN para classificação.

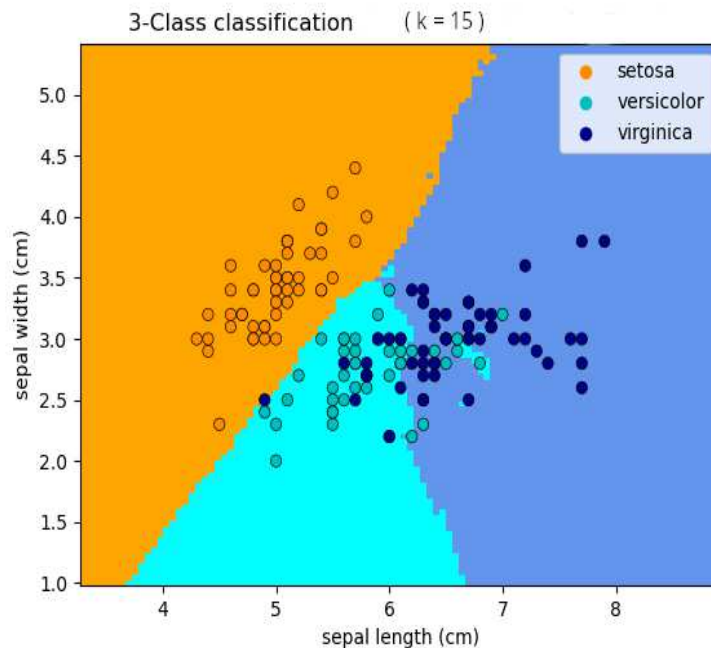


Figura 8 – Exemplo de classificação com *K*-nearest neighbors. (Fonte: Adaptado de [7])

3.4.3 Máquina de Vetor de Suporte

Support vector machines são algoritmos utilizados para classificar dados em duas categorias ou mais. Os dados são representados como pontos num espaço n-dimensional, onde o número n depende das características do *dataset*.

A SVM então define um hiperplano que pode ser em duas, três ou mais dimensões para separar os dados. Como há vários hiperplanos possíveis, a SVM tenta achar o melhor hiperplano maximizando as distâncias/margens entre os dados das classes definidas. Os pontos que estão exatamente sobre a margem são chamados de vetores de suporte [52].

A figura 9 mostra um exemplo das margens de uma SVM. Os pontos sob a linha tracejada são os vetores de suporte. A linha contínua é a linha de separação das classes.

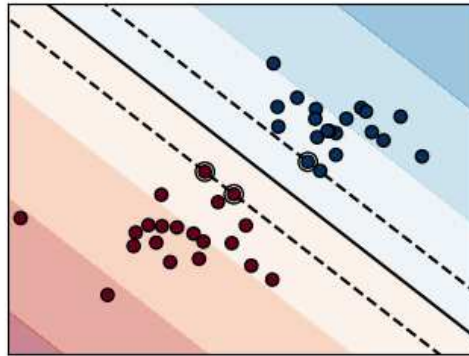


Figura 9 – Exemplo de margens de uma SVM. Os pontos sob as linhas tracejadas são os vetores de suporte. (Fonte: Adaptado de [8])

Uma *One-class SVM* é um conhecido algoritmo para detecção de anomalias, nesse caso, sabendo disso, a ideia de detecção de anomalia se encaixa bem para o problema se levado em consideração que os movimentos corretos terão maior coerência do que os dados incorretos. Além disso, é uma técnica já utilizada na literatura em [18].

Alguns dados não são separados linearmente no espaço n-dimensional, mas são separados em um espaço de maior dimensão, para isso é possível utilizar diferentes kernels. Foi utilizada a implementação do sklearn do *One-class SVM* [53] e desenvolvido em linguagem *Python* [48] utilizando o *Jupyter Notebook* [49]. O kernel escolhido foi o RBF (*Radius bases function*) que basicamente delimita o hiperplano ao redor de um raio como demonstrado no exemplo da figura 10.

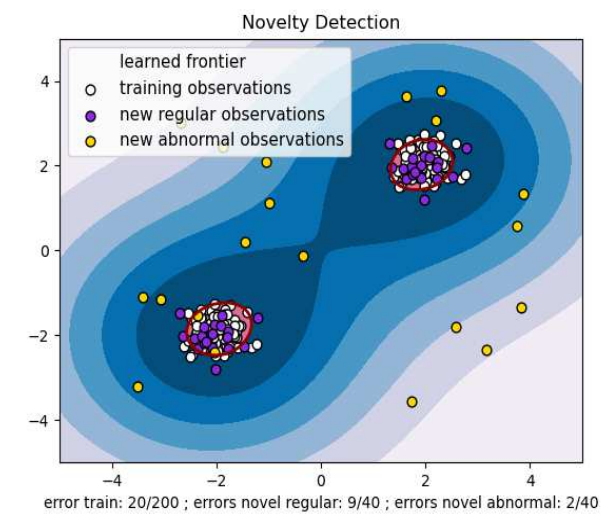


Figura 10 – Detecção de anomalia com *One-class SVM* com kernel RBF. (Fonte: [9])

3.5 Métricas de avaliação

As métricas de avaliação são importantes métodos para se obter informações do desempenho do modelo de aprendizado de máquina. Nesta seção serão discutidos os principais conceitos e métricas para avaliar um modelo de classificação.

3.5.1 Matriz de confusão

As métricas normalmente se baseiam em conceitos que são bem explicados através de uma matriz de confusão. A matriz de confusão é uma tabela que relaciona os erros e acertos do modelo de aprendizado de máquina. Através da matriz de confusão é possível comparar o resultado predito com o resultado esperado [54].

Através dessa comparação é possível identificar termos como: Verdadeiros Positivos (VP), Falsos Negativos (FN), Falsos Positivos (FP) e Verdadeiros Negativos (VN). Os conceitos precisam ser entendidos dentro do contexto onde existe uma classe correta (positiva) e uma classe incorreta (negativa) nos dados que serão preditos. Na tabela 2 é possível ver um exemplo de uma matriz de confusão.

Tabela 2 – Matriz de confusão

		Predito	
		Positivo	Negativo
Real	Positivo	Verdadeiro Positivo (VP)	Falso Negativo (FN)
	Negativo	Falso Positivo (FP)	Verdadeiro Negativo (VN)

São chamados de verdadeiros positivos os dados de classe positiva que obtiveram acerto na predição feita pelo modelo. Os falsos negativos são os dados da classe positiva que foram classificados como negativos pelo modelo. Os falsos positivos são os dados que o modelo classificou como positivos, mas são dados da classe negativa. Os verdadeiros negativos são dados da classe negativa que o modelo classificou corretamente como negativos.

Depois de se obter os termos da matriz de confusão é possível calcular algumas métricas que serão explicadas a seguir.

3.5.2 Acurácia

A acurácia é uma métrica que indica o desempenho geral do modelo de aprendizado de máquina. Essa métrica calcula, entre todas as classificações realizadas, quantas o modelo acertou a predição. A acurácia pode ser calculada através da equação 3.1:

$$Acurácia = \frac{VP + VN}{VP + VN + FP + FN} \quad (3.1)$$

3.5.3 Precisão

A precisão calcula quantas classificações corretas o modelo fez de classes que realmente eram positivas. A precisão pode ser calculada a partir da equação 3.2:

$$Precisão = \frac{VP}{VP + FP} \quad (3.2)$$

3.5.4 Recall-Score ou sensibilidade

O *recall-score* ou sensibilidade é uma métrica que calcula as classificações corretas de classes positivas dentre todas os dados que eram esperados como positivos. O *recall-score* pode ser calculado pela equação 3.3:

$$recall = \frac{VP}{VP + FN} \quad (3.3)$$

3.5.5 F-Score

O F-Score é uma métrica que une a precisão e o *recall-score*. Essa métrica é uma média harmônica entre a precisão e *recall-score*, sendo uma maneira de observar as duas métricas em uma só. O F-Score pode ser calculado pela equação 3.4:

$$F_{score} = 2 * \frac{(PRECISÃO) * (RECALL)}{(PRECISÃO) + (RECALL)} \quad (3.4)$$

3.6 Validação Cruzada

Quando se trabalha com modelos preditivos supervisionados, existem alguns problemas que devem ser levados em consideração. Um dos problemas principais é a dificuldade dos modelos em fazer uma boa generalização para dados que nunca foram vistos pelo modelo, a isso se dá o nome de *overfitting* [55].

Para solucionar o problema de *overfitting* nos modelos supervisionados existem algumas técnicas como *Single hold-out*, *Leave-one-out* e *K-fold*. Para este trabalho foi escolhida a técnica de validação cruzada chamada *K-fold*.

3.6.1 Validação cruzada com *K-fold*

Na validação cruzada *K-fold*, o *dataset* de treino é particionado em k partes de aproximadamente o mesmo tamanho, cada parte da subdivisão gera um *sub-dataset* que é

chamado de *fold*. Cada parte da divisão é então submetida ao modelo após o treino para fornecer uma validação. A divisão do *dataset* nessa técnica é feita de forma randômica, os dados separados para validação não são repostos. Dessa forma, cada parte da divisão tem sua chance de servir ao modelo como *set* de validação. [10]

A figura 11 exemplifica o processo de um K -*fold* de 10 partes. O *dataset* é dividido em 10 partes iguais, a primeira parte é separada como *set* de validação, as outras nove partes servem ao modelo como *set* de treino. Na segunda rodada, a segunda parte vai servir ao modelo como *set* de validação, e as outras nove como *set* de treino, e assim por diante até finalizar as 10 rodadas.

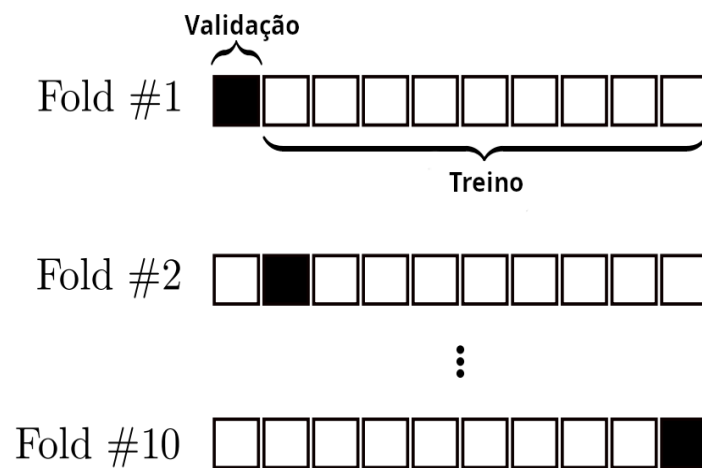


Figura 11 – Exemplo de validação cruzada com 10 -*fold*. (Fonte: Adaptado de [10])

4 MATERIAIS E MÉTODOS

Na figura 12 é possível verificar todo o processo metodológico, utilizado neste trabalho, para obter um modelo de aprendizado de máquina capaz de identificar se o movimento realizado pela criança está correto ou não.

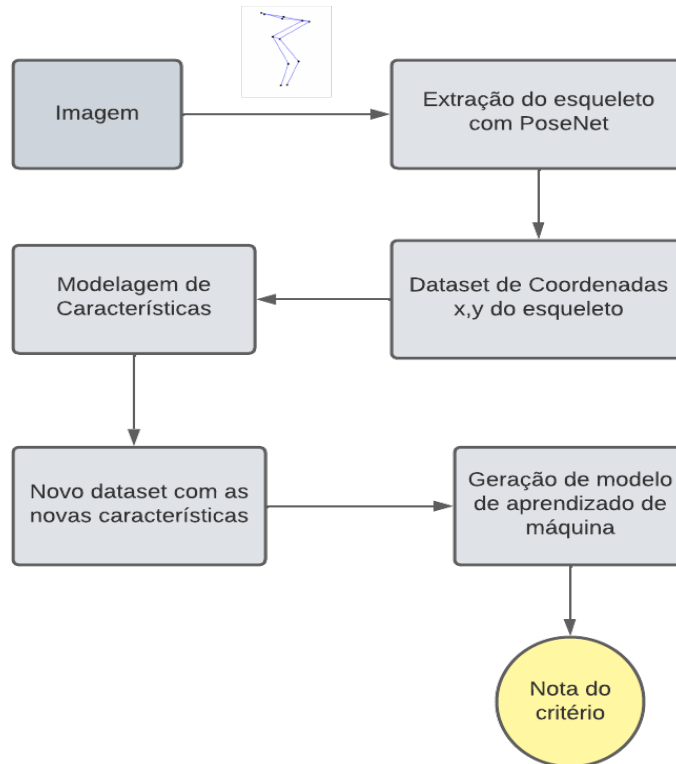


Figura 12 – Diagrama dos processos para obtenção do modelo. (Fonte: próprio autor)

4.1 *Dataset*

Para formação do *dataset* inicial foram utilizados os vídeos gravados para o trabalho de Spinosa [32]. No trabalho de Spinosa foram gravados vídeos de 131 crianças entre 3 e 11 anos de idade executando os movimentos de salto e arremesso, porém, para este trabalho foram utilizados apenas os vídeos da habilidade de salto, tendo em vista que o objetivo do trabalho é avaliar apenas o primeiro critério da habilidade de salto.

Os vídeos foram gravados com consentimento dos pais e concordância em utilização para trabalhos futuros. Os vídeos foram disponibilizados na plataforma Google Drive¹,

¹ <<https://www.google.com/intl/pt-BR/drive/>>

com um número de identificação para cada vídeo. Foram gravadas duas tentativas válidas para cada criança.

Os vídeos gravados para o trabalho de Spinosa foram submetidos a dois avaliadores que, avaliando os vídeos, atribuíram uma nota para cada critério da habilidade segundo o TGMD-3. Quando o movimento é executado corretamente, o participante recebeu nota 1, já quando o movimento não atende ao critério, recebe nota 0. [2]

Para este trabalho foram selecionados os vídeos de salto em que houve concordância de nota entre os avaliadores, ou seja, os vídeos em que os avaliadores atribuíram a mesma nota para o primeiro critério de salto, que é o critério abordado neste trabalho.

Foram encontrados 175 vídeos de salto que receberam nota 1, e mais de 175 vídeos de salto que receberam nota 0, portanto, para ter um balanceamento no *dataset*, foram selecionados randomicamente 175 vídeos que tiveram nota 0, assim, tendo um total de 350 vídeos no *dataset*.

Para cada um dos 350 vídeos selecionados, foram retiradas imagens em extensão PNG do momento em que a criança se prepara para o salto, atendendo ao momento que é avaliado no primeiro critério da habilidade de salto do TGMD-3. Esse processo foi longo, pois foi necessário abrir cada um dos 350 vídeos e extrair uma imagem do momento de preparação para o salto.

O resultado foi uma coleção de 350 imagens que refletem o momento de avaliação do primeiro critério da habilidade de salto, sendo 175 imagens avaliadas com nota 1 e 175 imagens avaliadas com nota 0.

4.2 Extração do esqueleto com *PoseNet*

A partir das imagens extraídas foi possível montar um programa simples em *JavaScript* que ao receber uma imagem em RGB executa o modelo *PoseNet*. Com a entrada de uma ou mais imagens, obtém-se a saída dos pontos-chave das articulações do corpo humano. O *PoseNet* é capaz de fornecer um esqueleto 2d do corpo humano como mostra a figura 13, assim gerando uma coordenada horizontal e outra vertical para cada ponto de articulação, os pontos que são identificados pelo *PoseNet* tensorflow são: olhos, nariz, boca, orelhas, ombros, cotovelos, pulsos, quadris, joelhos e tornozelos. [39]

Para este trabalho foram descartados olhos, nariz, boca e orelhas, tendo em vista que não interferem de forma alguma na execução das habilidades do TGMD-3, sendo assim não são nem objetos de avaliação no teste [2].

Sobraram então seis articulações que foram levadas em consideração, são elas: ombros, cotovelos, pulsos, quadris, joelhos e tornozelos, mas precisam ser consideradas tanto as articulações do lado direito do corpo humano quanto as articulações do lado

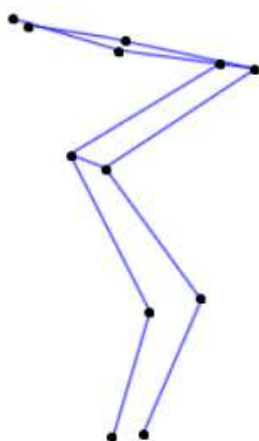


Figura 13 – Exemplo de esqueleto do corpo de uma criança extraído com *PoseNet*. (Fonte: Próprio autor)

esquerdo, então no total foram extraídas 12 articulações, 6 para o lado direito e 6 para o lado esquerdo do corpo.

Assim, formou-se um *dataset* inicial de todas as articulações passíveis de avaliação em uma habilidade do teste TGMD-3, e como para cada ponto de articulação são fornecidas duas coordenadas (horizontal e vertical), totalizou-se 24 coordenadas que representam o esqueleto das crianças nas imagens.

As 24 coordenadas, mais a nota de avaliação dos vídeos, formaram o *dataset* inicial para o trabalho. Sendo a nota o *target* a ser predito.

4.3 Seleção de características

A partir do *dataset* inicial e do descritivo do primeiro critério da habilidade de salto do TGMD-3, foi possível modelar uma estratégia para gerar novas características que representem melhor o critério de preparação para o salto.

No descritivo do primeiro critério da habilidade de salto, como mostra a figura 2, é informado que no momento da preparação para o salto os braços devem estar estendidos atrás do corpo e os joelhos devem estar flexionados [2]. Assim é possível extrair duas informações que devem ser avaliadas no primeiro critério da habilidade de salto: braços atrás do corpo e joelhos flexionados.

A partir dessas informações foi possível pensar em formas de abstrair características que representem melhor o problema, ter apenas as coordenadas dos pontos de corpo humano não é muito adequado, pois a posição de cada articulação vai depender do tamanho da imagem e de onde o corpo se localiza na imagem.

As imagens retiradas dos vídeos não possuem o mesmo tamanho, isso implica que as coordenadas do esqueleto não são viáveis para treinar um modelo sem antes passar por uma seleção de características, pois as coordenadas de uma imagem podem ser completamente deslocadas com relação a outra. Além disso, há crianças que foram gravadas executando o movimento da direita para a esquerda, e há crianças que foram gravadas executando o movimento da esquerda para a direita. Só isso já inviabiliza a utilização do *dataset* com as 24 coordenadas.

Outro problema é que as crianças não possuem a mesma altura, então as distâncias podem variar. Foi necessário pensar em estratégias para gerar características que são proporcionais e representam o primeiro critério da habilidade de salto. Ao invés de utilizar as coordenadas brutas de cada articulação do *dataset* inicial, é possível pensar em coordenadas relativas que contém mais uniformidade na representação da pose da criança.

Como todas as crianças foram gravadas lateralmente para a execução dessa habilidade, pode-se considerar o ponto que representa o ombro como delimitador do que está a frente e do que está atrás do corpo da criança. Assim, para estabelecer se os braços estão estendidos para trás, foi calculada a coordenada horizontal do ombro menos a coordenada horizontal do pulso, dessa forma é possível gerar uma característica positiva para braços atrás do corpo e negativa para braços à frente do corpo.

O cálculo foi feito para o lado direito e para o lado esquerdo, resultando em duas novas características para o novo *dataset*. A fórmula para calcular um lado do corpo é definida pela equação 4.1:

$$B = S_x - W_x \quad (4.1)$$

Onde B representa a diferença entre a posição horizontal do ombro e a posição horizontal do pulso, S_x é a posição horizontal do ombro e W_x é a posição horizontal do pulso.

Na maioria dos vídeos, as crianças executaram o movimento da esquerda para a direita, porém, houve 18 vídeos em que as crianças executaram o movimento da direita para a esquerda. Nesses vídeos, a ordem da subtração foi invertida, fazendo a coordenada horizontal do pulso menos a coordenada horizontal do ombro. Com isso, foi possível manter a característica positiva quando os braços estão atrás do corpo.

Quando os joelhos estão flexionados, a distância entre o quadril e o tornozelo, no eixo vertical, precisa ser menor do que quando a criança está em postura ereta. Para saber a distância quadril-tornozelo quando a criança está em postura ereta, foram calculadas as distâncias euclidianas do quadril-jelho e joelho-tornozelo no plano de coordenadas pela fórmula derivada do teorema de Pitágoras, as duas distâncias somadas resultam na distância quadril-tornozelo da criança em postura ereta.

Em seguida foi feita a diferença entre a distância quadril-tornozelo da criança em postura ereta e a distância vertical quadril-tornozelo aferida na imagem. O cálculo foi feito para o lado direito e para o lado esquerdo, resultando em duas novas características para o novo *dataset*. Quanto maior o resultado da diferença nesse cálculo, maior a flexão dos joelhos com relação à posição ereta da criança. A equação 4.2 representa o cálculo realizado para essa característica:

$$B = U - HA_y. \quad (4.2)$$

Onde B representa a diferença entre a distância quadril-tornozelo da criança em postura ereta e a distância quadril-tornozelo aferida na imagem, U representa a distância do quadril-tornozelo da criança em postura ereta, e HA_y a distância vertical do quadril-tornozelo aferida na imagem.

Após esses passos, obteve-se como resultado 4 novas características selecionadas, duas representando se o braço está atrás do corpo (direito e esquerdo) e duas representando a flexão dos joelhos (direito e esquerdo). Essas 4 novas características envolvem apenas as características originais que realmente importam para a avaliação. O resultado foi um novo *dataset* com as 4 características selecionadas mais a nota recebida no critério como *target*. O *dataset* inicial com 24 coordenadas não foi utilizado na fase de treino.

4.4 Visualização dos dados

Para visualização dos dados foi utilizada a técnica de *Principal Component Analysis* (PCA) [56]. A visualização de todos os dados a partir das novas características ficou de acordo com a figura 14. É possível identificar dois conjuntos, um núcleo de pontos onde o movimento foi feito de forma correta (verde) e outro onde o movimento foi feito de forma incorreta (vermelho). Para o PCA, foi utilizado o novo *dataset* com as 4 características.

4.5 Separação do *dataset*

Foram escolhidos três algoritmos para indução de um modelo capaz de classificar as imagens de forma a dizer se o movimento foi executado corretamente ou não. Os três algoritmos selecionados foram *K-nearest neighbors*, *Random Forest* e *One-class SVM*.

A separação dos dados de treino e teste foi feita randomicamente pelo K-fold, 20% do *dataset* foi separado como *holdout*. Foi tomada a devida atenção para separar dados como *holdout* para avaliar a performance do modelo depois da fase de treino. Os dados de *holdout*, teste e treino foram separados na devida proporção entre as duas classes 0 e 1, dessa forma, mantendo o balanceamento do *dataset* durante todo o processo. Isso resultou

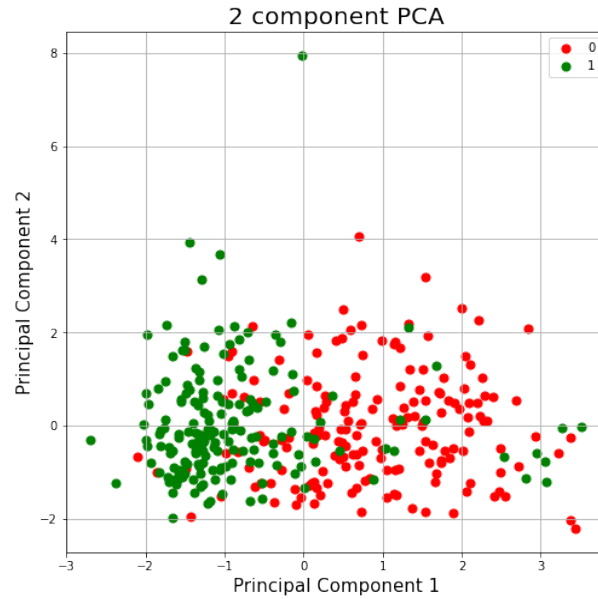


Figura 14 – Visualização dos dados com PCA de dois componentes. (Fonte: Próprio autor)

em 280 imagens para treino/teste e 70 imagens como *holdout*. Esse foi o procedimento para os algoritmos *K-nearest neighbors* e *Random Forest*.

Para o *One-class SVM* foi utilizado o *dataset* apenas com a classe 1, pois foi utilizada a técnica de detecção de anomalias, onde espera-se que as anomalias sejam todos os dados da classe 0. Os dados das 175 imagens com nota 1 foram divididos randomicamente em 80% para treino e 20% para teste, resultando em 140 imagens para treino e 35 imagens para teste. Das 175 imagens com classe 0 que ficaram fora do treino, foram selecionadas randomicamente 20%, gerando um *subdataset* de 70 imagens para testes, sendo 35 com classe 1 e 35 com classe 0.

5 RESULTADOS E DISCUSSÕES

Neste capítulo serão discutidos e apresentados os resultados dos experimentos para avaliação automática do primeiro critério da habilidade de salto do TGMD-3. O *dataset* com as 4 características selecionadas foi utilizado para a fase de treino, com o *target* a ser predito sendo a nota recebida no primeiro critério da habilidade de salto.

Na geração dos modelos com *K-nearest neighbors* e *Random Forest*, foi utilizado o *K-fold*, também do *sklearn* [46], para fazer a validação cruzada do *dataset*, assim, evitando o *overfitting* dos modelos. O *K-fold* utilizado dividiu o *dataset* em 5 partes. Foi utilizada a versão *Stratified K-fold*, dessa forma, os dados selecionados para a parte de teste são escolhidos de forma balanceada entre as duas classes 1 e 0.

Na implementação do *K-nearest neighbors* foi possível definir o número k de vizinhos que devem ser considerados para predizer a classe de um dado de entrada. O menor número de vizinhos que forneceu os melhores resultados foi 11. Além disso, é uma boa prática utilizar o número k de vizinhos como ímpar, assim forçando a decisão de uma classe se o dado de teste estiver exatamente na borda de separação entre duas classes.

Para o *K-nearest neighbors* o *K-fold* gerou 5 modelos de aprendizado de máquina, um para cada subdivisão da validação cruzada. Na tabela 3 são apresentados os resultados dos modelos gerados pelo *K-nearest neighbors* nas 5 rodadas.

Tabela 3 – Performance dos modelos do KNN de validação cruzada

Rodada	Acurácia	Precisão	<i>Recall-score</i>	F1-score
1	0.84	0.84	0.84	0.84
2	0.89	0.89	0.89	0.89
3	0.79	0.79	0.79	0.78
4	0.84	0.85	0.84	0.84
5	0.86	0.86	0.86	0.86

Fonte: Próprio autor.

Fazendo a média das métricas dos modelos do KNN, obteve-se uma acurácia de 0.84, uma precisão de 0.85, um *recall-score* e f1-score de 0.84.

O *sub-dataset* de *holdout*, ou seja, a parte separada do *dataset* que nunca foi vista pelos modelos na fase de treino, foi submetido separadamente à predição dos 5 modelos gerados pelo *K-fold*. Os resultados médios apresentados pelos modelos no *sub-dataset* de *holdout* foram de 0.85 de acurácia média, 0.90 de precisão média, 0.78 de *recall-score* médio e 0.84 de f1-score médio.

A figura 15 representa a matriz de confusão do melhor modelo gerado pelo *K-nearest neighbors* nos dados de *holdout*.

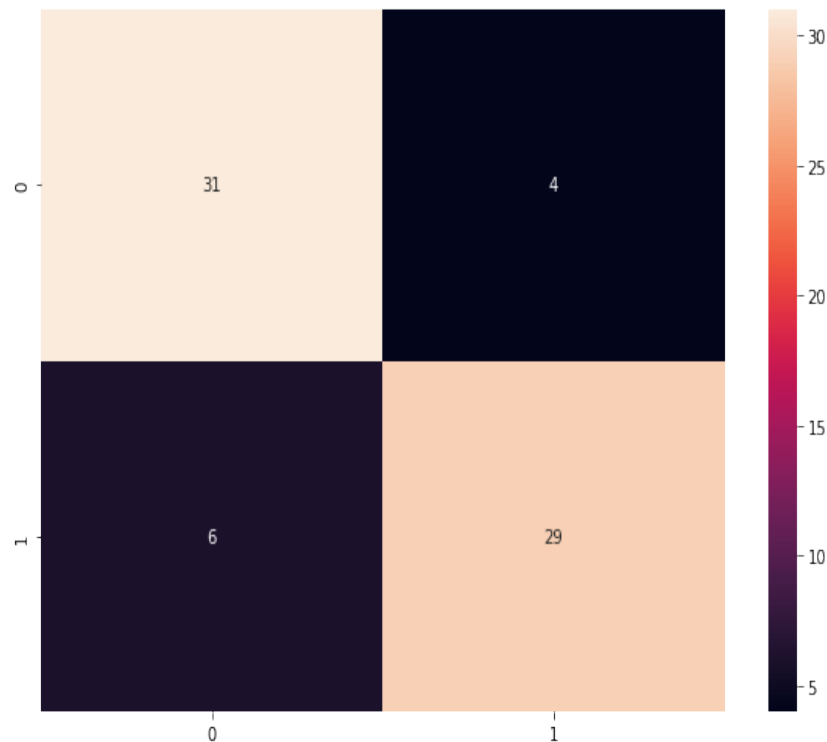


Figura 15 – Matriz de confusão do melhor modelo KNN nos dados de *holdout* (eixo vertical são os dados reais). (Fonte: próprio autor)

Na implementação do algoritmo *Random Forest* foi possível indicar o número de árvores de decisão a serem utilizadas, foram utilizadas 500 árvores de decisão no algoritmo, que é um valor acima do padrão, mas que forneceu bons resultados, e que é um número de árvores razoável para se criar um modelo. Para o *Random Forest* também foi utilizado o *K-fold* de 5 partes para fazer a validação cruzada e evitar *overfitting* dos modelos. Na tabela 4 são apresentados os resultados dos modelos gerados pelo *Random Forest* nas 5 rodadas.

Tabela 4 – Performance dos modelos do *Random Forest* de validação cruzada

Rodada	Acurácia	Precisão	<i>Recall-score</i>	F1-score
1	0.82	0.83	0.82	0.82
2	0.91	0.92	0.91	0.91
3	0.79	0.79	0.79	0.79
4	0.80	0.81	0.80	0.80
5	0.86	0.86	0.86	0.86

Fonte: Próprio autor.

Fazendo a média das métricas dos modelos do *Random Forest*, obteve-se uma acurácia de 0.84, uma precisão de 0.84, um *recall-score* de 0.84 e f1-score de 0.84.

O *sub-dataset* de *holdout* foi separadamente submetido à predição dos 5 modelos gerados pelo *K-fold*. Os resultados médios apresentados pelos modelos no *subdataset* de *holdout* foram de 0.93 de acurácia média, 0.89 de precisão média, 0.75 de *recall-score* médio e 0.81 de f1-score médio.

A figura 16 representa a matriz de confusão do melhor modelo gerado pelo *Random Forest* nos dados de *holdout*.

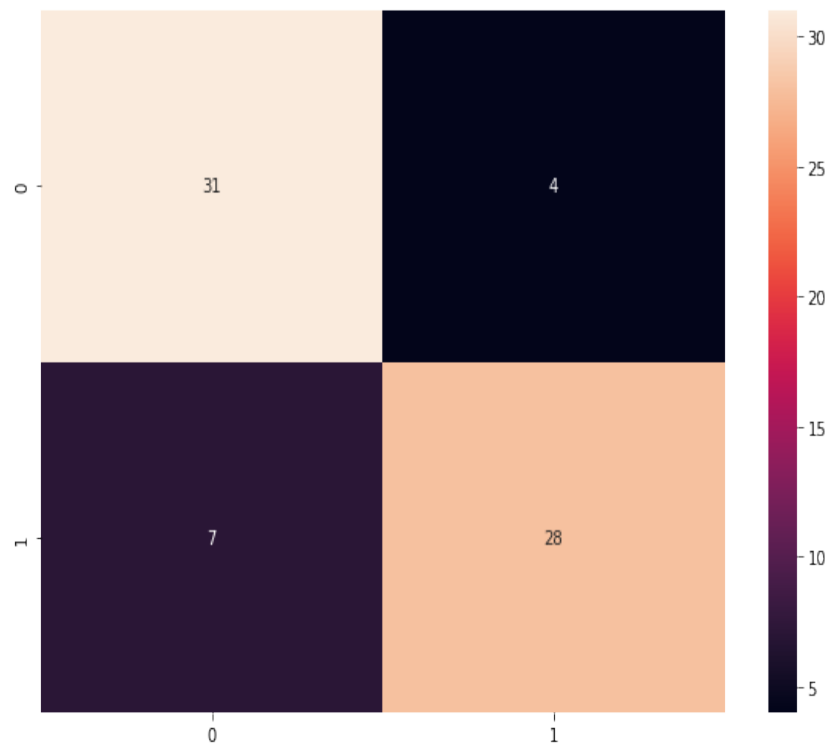


Figura 16 – Matriz de confusão do melhor modelo *Random Forest* nos dados de *holdout* (eixo vertical são os dados reais). (Fonte: próprio autor)

Na implementação da *One-class SVM* foram ajustados dois parâmetros. Primeiro foi definida a utilização do kernel RBF (*Radial basis function*) que é não-linear, isso implica na geração de um núcleo de pontos que será utilizado para definir se o movimento preparatório está correto ou não. Todas as anomalias fora do núcleo são considerados movimentos errados, por isso foram utilizados apenas os dados da classe 1 do *dataset*, condicionando o modelo a reconhecer tudo fora desse núcleo como movimento errado. Isso faz sentido se percebida a dispersão maior dos dados errados em comparação com os dados corretos na visualização dos dados na figura 14.

Então, foi definido que um total de 20% dos dados de treino devem ser considerados *outliers*, ou seja, devem estar fora do núcleo principal, 20% foi o número que gerou

melhores resultados na separação das classes para a *One-class SVM*.

Para o algoritmo *One-class SVM*, foram feitas 50 iterações, em cada iteração um novo *set* de treino e teste foi sorteado randomicamente na mesma proporção em que foram divididos inicialmente, ou seja, 80% para treino e 20% para testes, adicionando sempre aos dados de teste 20% de dados com classe 0 selecionados também randomicamente.

Para o modelo gerado pela *One-class SVM* a acurácia média foi de 0.77, precisão de 0.77, *recall-score* de 0.79 e o f1-score médio foi de 0.77.

Os resultados obtidos demonstram a possibilidade de geração de um modelo que é capaz de prever se o primeiro critério para a habilidade de salto foi executado de forma correta, atendendo ao objetivo geral do trabalho. Além disso, é possível indentificar o avanço em comparação com os trabalhos de [17] e [18].

Nos trabalhos de Suzuki, Amemiya e Sato (2021), o diagnóstico do movimento é feito de forma a mostrar visualmente ao avaliador onde possivelmente houve falha na execução do movimento, nesta dissertação foi apresentada uma abordagem de avaliação automática a partir de imagens que representam o primeiro critério, fornecendo a nota final para o critério automaticamente, porém, é necessário que sejam fornecidas imagens dos momentos certos que representam o primeiro critério, o que não se diferencia do trabalho de Suzuki, Amemiya e Sato (2021) no que diz respeito a essa dificuldade, já que em seu trabalho, também é necessário que o avaliador ativamente identifique os pontos do vídeo que representam os critérios.

Com relação ao trabalho de Bisi et al. (2017), essa dissertação se destaca no sentido de descartar a necessidade de equipamentos para que a avaliação seja feita. A concordância entre as avaliações de profissionais e a avaliação automática que Bisi et al. (2017) conseguiu em seu trabalho foi de 87% na média, o que se aproxima bastante dos resultados da proposta dessa dissertação, apesar da comparação ser difícil, pois no caso de Bisi, os movimentos completos são avaliados.

Além disso, este trabalho apresentou uma proposta totalmente inédita de como selecionar características para avaliar os critérios do TGMD-3, essas características selecionadas se provaram adequadas para treinar modelos que sejam capazes de identificar se o movimento foi executado de forma correta.

6 CONCLUSÃO

A aferição do desenvolvimento motor em crianças é de extrema importância, pois é fundamental tanto nas questões físicas como sociais da criança. Uma criança com déficit no desenvolvimento motor, certamente terá mais propensão a apresentar problemas de ordem física e cognitiva no seu período de maturação.

A implementação da avaliação automática do TGMD é um caminho muito natural, tendo em vista que o processo de avaliação é muito trabalhoso. As principais abordagens para solucionar o problema envolvem uso de equipamentos acoplados à criança durante a realização das habilidades ou fornecimento de um diagnóstico visual de possível incorreção na execução da habilidade.

O trabalho apresentou uma abordagem para solucionar o problema de avaliação automática do primeiro critério da habilidade de salto do TGMD-3. Foram extraídos os esqueletos das crianças com o *PoseNet* e apresentada uma modelagem inédita na seleção de características para representar o primeiro critério, ou seja, o movimento preparatório para o salto. Com a modelagem proposta foi possível induzir modelos capazes de prever se o movimento foi realizado de forma correta, cumprindo o objetivo principal com sucesso.

Foram gerados modelos de aprendizado de máquina utilizando os algoritmos *K-nearest neighbors*, *Random Forest* e *One-class SVM*. O modelo gerado pela *One-class SVM* obteve uma acurácia média de 77%, neste modelo foi utilizada a técnica de detecção de anomalias. Os modelos gerados pelo *K-nearest neighbors* e *Random Forest* obtiveram uma acurácia média de 84%.

Os resultados indicam um bom ajuste dos modelos e boa acurácia para avaliar se o primeiro critério da habilidade de salto do TGMD-3 foi executado corretamente. O trabalho se destaca, em comparação às outras abordagens, na simplificação dos processos envolvidos e direta inferência da nota do critério. Na abordagem proposta é descartada a utilização de sensores acoplados às crianças e equipamentos específicos.

Como trabalhos futuros seria interessante realizar o mesmo processo de modelagem de características para os critérios restantes da habilidade de salto, fornecendo uma visão geral da avaliação de uma habilidade completa, além da implementação da avaliação automática para as outras habilidades do teste TGMD-3.

Alguns desafios maiores podem ser encontrados, principalmente em habilidades que envolvem detecção de objetos, como as bolas utilizadas nas habilidades com bola, mas há perspectivas de novos estudos na área abordando a avaliação automática do desenvolvimento motor pelo teste TGMD-3.

Além disso é possível apontar como importante passo para um trabalho futuro, o reconhecimento automático dos pontos-chave nos vídeos das crianças, facilitando a identificação dos momentos do vídeo em que os critérios são avaliados.

REFERÊNCIAS

- [1] ULRICH, D. A.; WEBSTER, K. *TGMD - 3*. Disponível em: <<https://sites.google.com/view/tgmd-3/>>. Acesso em: 07/02/2023.
- [2] ULRICH, D. A. Introduction to the special section: Evaluation of the psychometric properties of the tgmd-3. *Journal of Motor Learning and Development*, Human Kinetics, Inc., Champaign IL, USA, v. 5, n. 1, p. 1 – 4, 2017. Disponível em: <<https://journals.humankinetics.com/view/journals/jmld/5/1/article-p1.xml>>.
- [3] MUNEA, T. L. et al. The progress of human pose estimation: A survey and taxonomy of models applied in 2d human pose estimation. *IEEE Access*, v. 8, p. 133330–133348, 2020.
- [4] GOOGLE CREATIVE LAB. *Real-time Human Pose Estimation in the Browser with TensorFlow.js*. 2018. Disponível em: <<https://blog.tensorflow.org/2018/05/real-time-human-pose-estimation-in.html>>. Acesso em: 07/02/2023.
- [5] JO, B.; KIM, S. Comparative analysis of openpose, posenet, and movenet models for pose estimation in mobile devices. *Traitement du Signal*, v. 39, n. 14, p. 119–124, 2022.
- [6] BIAU, G.; SCORNET, E. A random forest guided tour. *Test*, Springer, v. 25, p. 197–227, 2016.
- [7] SCIKIT-LEARN. *Nearest Neighbors Classification*. Disponível em: <https://scikit-learn.org/stable/auto_examples/neighbors/plot_classification.html>. Acesso em: 07/02/2023.
- [8] SCIKIT-LEARN. *SVM Margins Example*. Disponível em: <https://scikit-learn.org/stable/auto_examples/svm/plot_svm_margin.html#sphx-glr-auto-examples-svm-plot-svm-margin-py>. Acesso em: 22/05/2023.
- [9] LEARN, S. *One-class SVM with non-linear kernel (RBF)*. Disponível em: <https://scikit-learn.org/stable/auto_examples/svm/plot_oneclass.html>. Acesso em: 07/02/2023.
- [10] BERRAR, D. *Cross-Validation*. 2019.
- [11] GONZALEZ, S. L.; ALVAREZ, V.; NELSON, E. L. Do gross and fine motor skills differentially contribute to language outcomes? a systematic review. *Frontiers in Psychology*, v. 10, 2019. ISSN 1664-1078. Disponível em: <<https://www.frontiersin.org/articles/10.3389/fpsyg.2019.02670>>.
- [12] CLARK, J. E. From the beginning: A developmental perspective on movement and mobility. *Quest*, Routledge, v. 57, n. 1, p. 37–45, 2005. Disponível em: <<https://doi.org/10.1080/00336297.2005.10491841>>.
- [13] GALLAHUE, D. L.; OZMUN, J. C.; GOODWAY, J. D. *Compreendendo o desenvolvimento motor-: bebês, crianças, adolescentes e adultos*. [S.l.]: AMGH Editora, 2013.

- [14] MANOEL, E. D. J.; CONNOLLY, K. J. Variability and the development of skilled actions. *International Journal of Psychophysiology*, Elsevier, v. 19, n. 2, p. 129–147, 1995.
- [15] SANTOS, S.; DANTAS, L.; OLIVEIRA, J. A. d. Desenvolvimento motor de crianças, de idosos e de pessoas com transtornos da coordenação. *Rev Paul Educ Fís*, v. 18, n. 1, p. 33–44, 2004.
- [16] VALENTINI, N. C.; ZANELLA, L. W.; WEBSTER, E. K. Test of gross motor development—third edition: Establishing content and construct validity for brazilian children. *Journal of Motor Learning and Development*, Human Kinetics, Inc., Champaign IL, USA, v. 5, n. 1, p. 15 – 28, 2017. Disponível em: <<https://journals.humankinetics.com/view/journals/jmld/5/1/article-p15.xml>>.
- [17] BISI, M. C. et al. Objective assessment of movement competence in children using wearable sensors: An instrumented version of the tgmd-2 locomotor subtest. *Gait Posture*, v. 56, p. 42–48, 2017. ISSN 0966-6362. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0966636217301571>>.
- [18] SUZUKI, S.; AMEMIYA, Y.; SATO, M. Skeleton-based visualization of poor body movements in a child’s gross-motor assessment using convolutional auto-encoder. In: *2021 IEEE International Conference on Mechatronics (ICM)*. [S.l.: s.n.], 2021. p. 1–6.
- [19] AMEMIYA, Y.; SUZUKI, S.; SATOH, M. A support system for gross motor assessment of preschool children. In: *IECON 2018 - 44th Annual Conference of the IEEE Industrial Electronics Society*. [S.l.: s.n.], 2018. p. 4251–4256.
- [20] SUZUKI, S.; AMEMIYA, Y.; SATO, M. Enhancement of gross-motor action recognition for children by cnn with openpose. In: *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society*. [S.l.: s.n.], 2019. v. 1, p. 5382–5387.
- [21] SUZUKI, S.; AMEMIYA, Y.; SATO, M. Enhancement of child gross-motor action recognition by motional time-series images conversion. In: *2020 IEEE/SICE International Symposium on System Integration (SII)*. [S.l.: s.n.], 2020. p. 225–230.
- [22] ADOLPH, K. E.; ROBINSON, S. R. Motor development. In: _____. *Handbook of Child Psychology and Developmental Science*. John Wiley Sons, Ltd, 2015. cap. 4, p. 1–45. ISBN 9781118963418. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118963418.childpsy204>>.
- [23] DAVIS, E. E.; PITCHFORD, N. J.; LIMBACK, E. The interrelation between cognitive and motor development in typically developing children aged 4–11 years is underpinned by visual processing and fine manual control. *British Journal of Psychology*, v. 102, n. 3, p. 569–584, 2011. Disponível em: <<https://bpspsychub.onlinelibrary.wiley.com/doi/abs/10.1111/j.2044-8295.2011.02018.x>>.
- [24] SANTOS, M. C. S. et al. Uso da escala de desenvolvimento motor: uma revisão integrativa. *Revista CEFAC*, SciELO Brasil, v. 21, 2019.

- [25] VALENTINI, N. C.; RUDISILL, M. E. Motivational climate, motor-skill development, and perceived competence: Two studies of developmentally delayed kindergarten children. *Journal of Teaching in Physical Education*, Human Kinetics, Inc., Champaign IL, USA, v. 23, n. 3, p. 216 – 234, 2004. Disponível em: <<https://journals.humankinetics.com/view/journals/jtpe/23/3/article-p216.xml>>.
- [26] I SANTOS C. R. DOS, M.-P. J. M. *Teste de desenvolvimento motor para educação física escolar*. [S.l.]: CRV, 2017.
- [27] KIPHARD, E. J.; SCHILLING, V. F. Köperkoordinationstest für kinder ktk: Manual. Weinheim: Beltz Test, 1974.
- [28] NETO, F. R. Manual de avaliação motora. Porto Alegre, Artmed, 2002.
- [29] S. SUGDEN D., B. A. H. Movement assessment battery for children. 2nd ed. Henderson S, Sugden DA, Barnett A, 2007.
- [30] ULRICH, D. A. The test of gross motor development. Austin, 2000.
- [31] BURTON, A. W.; MILLER, D. E. *Movement skill assessment*. [S.l.]: Human Kinetics, 1998. 58 p.
- [32] SPINOSA, R. M. de O. *Demonstração Digital de Habilidades Motoras aplicada à Instrumentos de valiação do Desenvolvimento Motor*. Tese (Doutorado) — Universidade Estadual de Londrina - UEL, Londrina - PR, Brasil, 2019.
- [33] ISSARTEL, J. et al. A cross-validation study of the tgmd-2: The case of an adolescent population. *Journal of Science and Medicine in Sport*, v. 20, n. 5, p. 475–479, 2017. ISSN 1440-2440. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1440244016302134>>.
- [34] VALENTINI, N. C. et al. Test of gross motor development: Expert validity, confirmatory validity and internal consistence. *Revista Brasileira de Cineantropometria e Desempenho Humano*, v. 10, n. 4, p. 399–404, 2008.
- [35] VALENTINI, N. C.; ZANELLA, L. W.; WEBSTER, E. K. Test of gross motor development—third edition: Establishing content and construct validity for brazilian children. *Journal of Motor Learning and Development*, Human Kinetics, Inc., v. 5, n. 1, p. 15–28, 2017.
- [36] LIN, T.-Y. et al. Microsoft coco: Common objects in context. In: FLEET, D. et al. (Ed.). *Computer Vision – ECCV 2014*. Cham: Springer International Publishing, 2014. p. 740–755. ISBN 978-3-319-10602-1.
- [37] ANDRILUKA, M. et al. 2d human pose estimation: New benchmark and state of the art analysis. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2014.
- [38] FIERARU, M. et al. *Learning to Refine Human Pose Estimation*. arXiv, 2018. Disponível em: <<https://arxiv.org/abs/1804.07909>>.
- [39] PAPANDREOU, G. et al. *PersonLab: Person Pose Estimation and Instance Segmentation with a Bottom-Up, Part-Based, Geometric Embedding Model*. arXiv, 2018. Disponível em: <<https://arxiv.org/abs/1803.08225>>.

- [40] ALBAWI, S.; MOHAMMED, T. A.; AL-ZAWI, S. Understanding of a convolutional neural network. In: *2017 International Conference on Engineering and Technology (ICET)*. [S.l.: s.n.], 2017. p. 1–6.
- [41] ROY, V.; DESAI, A. Chapter 5 - a heuristic approach of web users decision-making using deep learning models. In: MISHRA, S. et al. (Ed.). *Cognitive Big Data Intelligence with a Metaheuristic Approach*. Academic Press, 2022, (Cognitive Data Science in Sustainable Computing). p. 117–130. ISBN 978-0-323-85117-6. Disponível em: <<https://www.sciencedirect.com/science/article/pii/B9780323851176000157>>.
- [42] JLIDI, N. et al. PTLHAR: PoseNet and transfer learning for human activities recognition based on body articulations. In: OSTEN, W.; NIKOLAEV, D. P. (Ed.). *Twelfth International Conference on Machine Vision (ICMV 2019)*. SPIE, 2020. v. 11433, p. 114330Q. Disponível em: <<https://doi.org/10.1117/12.2559567>>.
- [43] QUINLAN, J. R. Induction of decision trees. *Machine learning*, Springer, v. 1, p. 81–106, 1986.
- [44] KOTSIANTIS, S. B. Decision trees: a recent overview. *Artificial Intelligence Review*, Springer, v. 39, p. 261–283, 2013.
- [45] BREIMAN, L. Random forests. *Machine learning*, Springer, v. 45, p. 5–32, 2001.
- [46] PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, v. 12, p. 2825–2830, 2011.
- [47] SCIKIT-LEARN. *RandomForestClassifier*. Disponível em: <<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>>. Acesso em: 22/05/2023.
- [48] ROSSUM, G. V.; JR, F. L. D. *Python reference manual*. [S.l.]: Centrum voor Wiskunde en Informatica Amsterdam, 1995.
- [49] KLUYVER, T. et al. Jupyter notebooks – a publishing format for reproducible computational workflows. In: LOIZIDES, F.; SCHMIDT, B. (Ed.). *Positioning and Power in Academic Publishing: Players, Agents and Agendas*. [S.l.], 2016. p. 87 – 90.
- [50] LAROSE, D. T.; LAROSE, C. D. kx2010;nearest neighbor algorithm. In: _____. *Discovering Knowledge in Data: An Introduction to Data Mining*. [S.l.: s.n.], 2014. p. 149–164.
- [51] SCIKIT-LEARN. *KNeighborsClassifier*. Disponível em: <<https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html>>. Acesso em: 07/02/2023.
- [52] CORTES, C.; VAPNIK, V. Support-vector networks. *Machine learning*, Springer, v. 20, n. 3, p. 273–297, 1995.
- [53] SCIKIT-LEARN. *OneClassSVM*. Disponível em: <<https://scikit-learn.org/stable/modules/generated/sklearn.svm.OneClassSVM.html>>. Acesso em: 07/02/2023.
- [54] HANDELMAN, G. S. et al. Peering into the black box of artificial intelligence: evaluation metrics of machine learning methods. *American Journal of Roentgenology*, Am Roentgen Ray Soc, v. 212, n. 1, p. 38–43, 2019.

- [55] YING, X. An overview of overfitting and its solutions. In: IOP PUBLISHING.
Journal of physics: Conference series. [S.l.], 2019. v. 1168, p. 022022.
- [56] F.R.S., K. P. Liii. on lines and planes of closest fit to systems of points in space.
The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science,
Taylor Francis, v. 2, n. 11, p. 559–572, 1901.

TRABALHOS ACEITOS

Pilati Filho, E. L.; Brancher, J. D.; Spinosa, R. M. de O. **A machine learning model to automatic assessment of gross motor development in children using PoseNet** aceito na revista **Semina: Exact and Technological Sciences**, Universidade Estadual de Londrina, em 14 de setembro de 2023. (Qualis B4, 2017 - 2020)