

Universidade Federal de Ouro Preto - UFOP
Instituto Tecnológico Vale - ITV MI
Departamento de Computação - DECOM
Ciência da Computação

Projeto: Prognóstico de Falhas em Correias Transportadoras

Co-particionamento para Suporte à Implementação de Redes Neurais
Convulacionais

Niege Reis
Professor: Fernando Cortez Sica

Ouro Preto
19 de abril de 2024

Sumário

1	Introdução	1
2	Referencial teórico/revisão bibliográfica	2
2.1	Implementation of Fully-Pipelined CNN Inference Accelerator on FPGA and HBM2 Platform	2
2.2	Efficient Hardware Architectures for Accelerating Deep Neural Networks: Survey	2
2.3	An FPGA-Based Hardware Accelerator for CNNs Inference on Board Satellites: Benchmarking with Myriad 2-Based Solution for the CloudScout Case Study	2
2.4	Automatic Deployment of Convolutional Neural Networks on FPGA for Spaceborne Remote Sensing Application	2
3	Cenário do Projeto	3
3.1	Contextualização	3
3.2	Sensores e Coleta de Dados	3
3.3	Classificação por Rede Supervisionada	3
3.4	Implementação Inicial	4
3.5	Benefícios Esperados	4
4	Desenvolvimento	4
4.1	Algoritmo de Rede Supervisionada	4
4.2	Descrição Geral	4
4.3	Implementação	4
4.4	Treinamento e Avaliação	6
5	Resultados Preliminares	6
5.1	Avaliação do Desempenho	6
5.2	Métricas de Desempenho	6
5.3	Análise de Convergência	6
5.4	Discussão dos Resultados Preliminares	6
6	Conclusão e Trabalhos Futuros	7
6.1	Conclusões	7
6.2	Trabalhos Futuros	7
6.2.1	Implementação em FPGA	7
6.2.2	Otimização para Eficiência Energética	7
6.2.3	Expansão da Base de Dados	7
6.2.4	Validação em Ambiente Real	7
6.3	Considerações Finais	7

Lista de Figuras

1	Quadro comparativo.	3
---	-----------------------------	---

Lista de Tabelas

1	Resultados obtidos com o teste da rede.	6
---	---	---

Lista de Códigos Fonte

1	Implementação da rede.	5
---	--------------------------------	---

1 Introdução

O "Projeto Prognóstico de Falhas em Correias Transportadoras" representa uma iniciativa voltada para o avanço da manutenção preditiva, focalizando a antecipação de potenciais falhas em correias transportadoras. Em um cenário onde a integridade desses componentes é crítica para a operação eficiente, a implementação de técnicas avançadas de prognóstico assume um papel fundamental na redução de custos operacionais e na maximização da confiabilidade dos sistemas.

Nesse contexto, giroscópios e acelerômetros são empregados para classificar movimentos desde o instante inicial até o momento n . A análise desse fator de coesão proporciona uma base sólida para o desenvolvimento de modelos preditivos.

A classificação, elemento essencial neste projeto, é abordada por meio de uma rede supervisionada. Esta rede utiliza a função de ativação padrão, sendo submetida a testes, como movimentações para frente e paradas da correia. A implementação é feita utilizando o framework TensorFlow projetado para o ESP32 e incorporando funcionalidades do Keras. A avaliação dos resultados obtidos nessa fase inicial direciona as escolhas subsequentes, preparando o terreno para a posterior implementação no ambiente C e após isso o algoritmo em verilog para FPGA.

Este projeto, ao integrar o prognóstico de falhas em correias transportadoras com a análise avançada de dados dos sensores, almeja não apenas otimizar a confiabilidade operacional, mas também pavimentar o caminho para abordagens inovadoras na manutenção preditiva industrial.

2 Referencial teórico/revisão bibliográfica

A revisão bibliográfica é essencial para contextualizar o projeto, fornecendo uma base sólida e entendimento dos conceitos e tecnologias relevantes. Os seguintes artigos foram estudados para embasar o desenvolvimento do sistema de prognóstico de falhas em correias transportadoras:

2.1 Implementation of Fully-Pipelined CNN Inference Accelerator on FPGA and HBM2 Platform

Este artigo aborda a implementação eficiente de um acelerador de inferência para Redes Neurais Convolucionais (CNN) em FPGA e plataforma HBM2. A ênfase na total pipelinação destaca a importância da otimização de hardware para a aceleração de redes neurais, fornecendo insights valiosos para a implementação do acelerador proposto neste projeto.

2.2 Efficient Hardware Architectures for Accelerating Deep Neural Networks: Survey

Esta pesquisa realiza uma revisão abrangente das arquiteturas de hardware eficientes para acelerar Redes Neurais Profundas (DNN). Explora diversas abordagens e técnicas, fornecendo uma visão geral do campo de aceleração de redes neurais. As informações desse artigo contribuem para a compreensão aprofundada da eficiência de hardware em implementações de redes neurais.

2.3 An FPGA-Based Hardware Accelerator for CNNs Inference on Board Satellites: Benchmarking with Myriad 2-Based Solution for the CloudScout Case Study

Este estudo focaliza a implementação de um acelerador de hardware baseado em FPGA para inferência de CNNs em satélites. A comparação com uma solução baseada em Myriad 2 destaca a eficácia de aceleradores FPGA em cenários espaciais. A análise de benchmarks oferece insights cruciais para a implementação de aceleradores em hardware para o projeto de prognóstico de falhas em correias transportadoras.

2.4 Automatic Deployment of Convolutional Neural Networks on FPGA for Spaceborne Remote Sensing Application

O artigo concentra-se na implantação automática de Redes Neurais Convolucionais em FPGAs para aplicações de sensoriamento remoto em espaçonaves. O estudo aborda desafios específicos relacionados à implementação eficiente de CNNs em FPGAs, contribuindo com conhecimentos cruciais para o projeto em questão.

Esses artigos fornecem uma base sólida para a compreensão das melhores práticas, desafios e inovações no campo de aceleração de redes neurais em hardware, orientando as escolhas metodológicas e estratégias adotadas no desenvolvimento do sistema de prognóstico de falhas em correias transportadoras.

Aspecto/Artigo	"Fully-Pipelined CNN Inference Accelerator on FPGA and HBM2 Platform"	"Efficient Hardware Architectures for Accelerating Deep Neural Networks: Survey"	"FPGA-Based Hardware Accelerator for CNNs Inference on Board Satellites"	"Automatic Deployment of Convolutional Neural Networks on FPGA for Spaceborne Remote Sensing Application"
Arquitetura	FPGA, HBM2	Diversas arquiteturas exploradas em pesquisas	FPGA	FPGA
Foco Principal	Inferência CNN totalmente pipelinizada	Eficiência de hardware para aceleração de DNNs	Acelerador FPGA para inferência CNNs em satélites	Implantação automática de CNNs em FPGAs para sensoriamento remoto em espaçonaves
Benchmarking	-	-	Comparação com solução Myriad 2	Exploração de eficiência para aplicações de sensoriamento remoto espacial
Aplicabilidade	-	Amplas aplicações de aceleração de DNNs	Satélites, espaço	Sensoriamento remoto espacial
Resultados-Chave	Implementação eficiente para FPGA e HBM2, pipelinização completa	Revisão abrangente de arquiteturas eficientes	Comparação com Myriad 2, ênfase em espaço	Implantação automática bem-sucedida de CNNs em FPGA para aplicações de sensoriamento remoto em espaçonaves
Desafios	-	Diversidade de arquiteturas, otimização de hardware	Desafios espaciais, eficiência energética em satélites	Implantação eficiente de CNNs em ambientes espaciais, otimização para sensoriamento remoto
Contribuições	Pipelinização eficiente, implementação específica para FPGA e HBM2	Visão abrangente do estado-da-arte em arquiteturas de aceleração para DNNs	Benchmarking espacial, enfatiza aceleradores FPGA	Implantação automática bem-sucedida em FPGA para aplicações de sensoriamento remoto espacial

Figura 1: Quadro comparativo.

Quadro Comparativo: Algoritmos de Aceleração para Redes Neurais em Hardware.

Este quadro comparativo resume os aspectos principais dos algoritmos e implementações presentes nos artigos estudados, destacando suas arquiteturas, enfoques, aplicabilidades, resultados, desafios enfrentados e contribuições para o campo de aceleração de redes neurais em hardware.

3 Cenário do Projeto

O projeto de Prognóstico de Falhas em Correias Transportadoras tem como propósito desenvolver uma solução para antecipar e prevenir falhas em sistemas de transporte do minério por correias. O foco principal reside na utilização de sensores, incluindo giroscópios e acelerômetros, para classificar movimentos desde o momento inicial até um instante determinado.

3.1 Contextualização

O setor de transporte por correias desempenha um papel essencial na Vale, sendo crucial para a transportação de minério. No entanto, a ocorrência de falhas nas correias transportadoras pode acarretar em custos operacionais elevados, paradas não programadas e potenciais danos ao equipamento.

3.2 Sensores e Coleta de Dados

Os sensores empregados, giroscópios e acelerômetros, desempenham um papel fundamental na coleta de dados em tempo real relacionados ao desempenho da correia transportadora.

3.3 Classificação por Rede Supervisionada

O processo de classificação das informações é conduzido por meio de uma rede neural supervisionada, que faz uso da função de ativação padrão. A avaliação dessa rede é realizada mediante comportamentos específicos, integrando um framework TensorFlow personalizado para o ESP32 e incorporando funcionalidades do Keras para otimização e flexibilidade no desenvolvimento.

3.4 Implementação Inicial

A implementação inicial foi feita utilizando o framework TensorFlow. Os resultados obtidos são minuciosamente avaliados para identificar a abordagem mais eficaz. Posteriormente, há a implementação no ambiente C, visando a otimização e integração mais ampla com sistemas operacionais, como ESP32.

3.5 Benefícios Esperados

- Proatividade na Manutenção: Antecipação de falhas possibilita intervenções proativas, reduzindo o tempo de inatividade e os custos associados à manutenção corretiva.
- Otimização Operacional: Melhoria na eficiência operacional ao evitar paradas não programadas e otimizar estratégias de manutenção.
- Prolongamento da Vida Útil: Identificação precoce de potenciais falhas contribui para a extensão da vida útil dos equipamentos.

Este cenário estabelece as bases para um projeto abrangente de prognóstico de falhas, integrando tecnologias de sensoriamento avançado, aprendizado de máquina e otimização de implementação. O foco na proatividade e eficiência operacional destaca a relevância e impacto positivo esperado desse projeto no setor de transporte por correias.

4 Desenvolvimento

4.1 Algoritmo de Rede Supervisionada

Nesta seção, é descrito a implementação do algoritmo de Rede Neural Supervisionada para o Prognóstico de Falhas em Correias Transportadoras.

4.2 Descrição Geral

O algoritmo adotado é uma Rede Neural Supervisionada, projetada para classificar os padrões de movimentação obtidos a partir de giroscópios e acelerômetros. Essa abordagem é crucial para a identificação precoce de potenciais falhas nas correias transportadoras.

4.3 Implementação

O código de implementação tem como objetivo desenvolver um modelo de Rede Neural Supervisionada para prever se uma correia transportadora está em movimento ou parada, utilizando dados coletados por sensores, como giroscópios e acelerômetros. O processo é dividido em etapas fundamentais:

- Carregamento e Pré-processamento dos Dados: os dados de movimento são carregados de uma planilha, sendo preparados para o treinamento ao separar características e definir a variável alvo.
- Divisão dos Dados: os dados são divididos em conjuntos de treinamento e teste para avaliação do modelo.
- Normalização dos Dados: as características são normalizadas para assegurar consistência na escala dos dados.
- Construção da Rede Neural: uma rede neural é construída com camadas densas, a última utilizando ativação sigmoid para classificação binária.
- Compilação do Modelo: o modelo é compilado com otimizador Adam, função de perda binary-crossentropy e a métrica accuracy para avaliação do desempenho.
- Treinamento do Modelo: o modelo é treinado usando dados de treinamento, ajustando-se para otimizar a função de perda.
- Avaliação do Modelo: a precisão do modelo é avaliada utilizando dados de teste para medir sua capacidade de generalização.

- Exibição da Precisão: a precisão alcançada durante o teste é exibida no console.

```
1 import pandas as pd
2 from sklearn.model_selection import train_test_split
3 from sklearn.preprocessing import StandardScaler
4 from tensorflow import keras
5 from keras import layers
6 from sklearn.metrics import accuracy_score, confusion_matrix
7
8 dados = pd.read_excel('caminho_da_planilha')
9
10 X = dados.drop('Movimento ', axis=1)
11 y = dados['Movimento ']
12
13 tamanho_teste = qtd_amostras
14
15 X_treino, X_teste, y_treino, y_teste = train_test_split(X, y, test_size=
    tamanho_teste, shuffle=False)
16
17 scaler = StandardScaler()
18 X_treino = scaler.fit_transform(X_treino)
19 X_teste = scaler.transform(X_teste)
20
21
22 modelo = keras.Sequential([
23     layers.Dense(128, activation='relu', input_shape=(X_treino.shape[1],)),
24     layers.Dropout(0.5),
25     layers.Dense(64, activation='relu'),
26     layers.Dropout(0.3),
27     layers.Dense(32, activation='relu'),
28     layers.Dense(16, activation='relu'),
29     layers.Dense(1, activation='sigmoid')
30 ])
31
32 optimizer = keras.optimizers.Adam(learning_rate=0.005)
33 modelo.compile(optimizer=optimizer, loss='binary_crossentropy', metrics=['
    accuracy'])
34
35 modelo.fit(X_treino, y_treino, epochs=50, batch_size=32, validation_split=0.2)
36
37 probabilidades = modelo.predict(X_teste)
38 y_pred = (probabilidades > 0.5).astype("int32")
39
40 print(f'Acuracia: {accuracy_score(y_teste, y_pred)}')
41
42 previsoes = modelo.predict(X_teste)
43
44 for i in range(len(previsoes)):
45     if previsoes[i] > 0.5:
46         if y_teste.iloc[i] == 1:
47             print(f"Amostra {i + 1}: Correto! Em movimento!")
48         else:
49             print(f"Amostra {i + 1}: Incorreto! Parado!")
50     else:
51         if y_teste.iloc[i] == 0:
52             print(f"Amostra {i + 1}: Correto! Parado!")
53         else:
54             print(f"Amostra {i + 1}: Incorreto! Em movimento!")
```

Código 1: Implementação da rede.

4.4 Treinamento e Avaliação

Os dados coletados dos sensores foram submetidos a um processo de pré-processamento e divididos em conjuntos de treinamento e teste para a implementação do modelo de Rede Neural Supervisionada. O treinamento do modelo foi conduzido utilizando o otimizador 'adam', a função de perda 'binary-crossentropy' e a métrica 'accuracy' para avaliação do desempenho durante o processo de aprendizado.

5 Resultados Preliminares

Nesta seção, é apresentado os resultados preliminares obtidos durante a fase inicial do projeto de Prognóstico de Falhas em Correias Transportadoras, com ênfase na aplicação do algoritmo de Rede Neural Supervisionada.

5.1 Avaliação do Desempenho

Os resultados preliminares indicam um desempenho promissor do modelo de Rede Neural Supervisionada na tarefa de classificação de padrões de movimentação. A acurácia alcançada nos conjuntos de teste demonstra a capacidade do algoritmo em aprender e a partir dos dados fornecidos pelos giroscópios e acelerômetros, durante a realização de testes foi evidente que quanto maior a quantidade de dados existentes na planilha maior foi a acurácia do algoritmo.

5.2 Métricas de Desempenho

Quantidade de dados(amstras)	Acurácia
100	0.60
200	0.78
300	0.72
400	0.69
500	0.75
600	0.73
700	0.56
800	0.69
900	0.68
1000	1.0
5000	1.0

Tabela 1: Resultados obtidos com o teste da rede.

5.3 Análise de Convergência

Durante o treinamento, observou-se a convergência do modelo após um número adequado de iterações. A análise da função de perda ao longo do tempo evidenciou a estabilização do aprendizado a partir de teste com amostra de tamanho 1000, sugerindo que o algoritmo foi capaz de capturar efetivamente os padrões relevantes nos dados.

5.4 Discussão dos Resultados Preliminares

Os resultados preliminares apontam para a eficácia do algoritmo na classificação de padrões de movimentação. No entanto, é importante destacar que esta análise está em estágio inicial. Subsequentes iterações de treinamento e validação serão conduzidas para o refinamento do modelo e validação mais robusta dos resultados.

6 Conclusão e Trabalhos Futuros

6.1 Conclusões

O desenvolvimento e avaliação preliminar do algoritmo de Rede Neural Supervisionada para o Prognóstico de Falhas em Correias Transportadoras apresentaram resultados encorajadores. A capacidade do modelo em classificar padrões de movimentação, utilizando dados de giroscópios e acelerômetros, sugere seu potencial como uma ferramenta valiosa para detecção precoce de falhas no sistema.

A análise do desempenho, convergência do treinamento e visualização da arquitetura proporcionam insights valiosos sobre o comportamento do modelo. A acurácia alcançada nos testes preliminares valida a eficácia da abordagem de aprendizado supervisionado.

6.2 Trabalhos Futuros

Considerando o escopo do projeto e visando aprimorar ainda mais sua implementação, destacamos os seguintes pontos para trabalhos futuros:

6.2.1 Implementação em FPGA

O próximo passo crucial consistirá na migração bem-sucedida do modelo treinado para um ambiente de FPGA. Isso envolverá a adaptação da arquitetura para atender às especificidades do hardware reconfigurável, explorando eficientemente suas capacidades de processamento paralelo.

6.2.2 Otimização para Eficiência Energética

A eficiência energética é um fator crítico, especialmente em dispositivos embarcados. Trabalhos futuros incluirão a otimização do modelo para garantir um consumo de energia adequado, alinhado com as restrições de dispositivos de borda.

6.2.3 Expansão da Base de Dados

Para melhorar a capacidade de generalização do modelo, será necessário expandir a base de dados de treinamento, incorporando uma variedade mais ampla de cenários e condições operacionais.

6.2.4 Validação em Ambiente Real

A validação do algoritmo em condições reais de operação, envolvendo a coleta de dados em campo, será essencial para verificar a robustez do modelo em situações práticas.

6.3 Considerações Finais

As conclusões deste estágio inicial são promissoras, delineando um caminho claro para a implementação em FPGA e otimização contínua do algoritmo. O compromisso com uma abordagem iterativa e a adaptação às demandas específicas do ambiente contribuirá para o sucesso contínuo deste projeto de Prognóstico de Falhas em Correias Transportadoras.