



ANDREY GUSTAVO DE SOUZA

**SISTEMA DE IDENTIFICAÇÃO DE CONDUTORES BASEADO
EM TÉCNICAS DE INTELIGÊNCIA COMPUTACIONAL**

LAVRAS – MG

2018

ANDREY GUSTAVO DE SOUZA

**SISTEMA DE IDENTIFICAÇÃO DE CONDUTORES BASEADO EM TÉCNICAS DE
INTELIGÊNCIA COMPUTACIONAL**

Projeto de pesquisa apresentado à Universidade Federal de Lavras, como parte das exigências do Programa de Pós-Graduação em Engenharia de Sistemas e Automação para a obtenção do título de Mestre.

Prof. Dr. Wilian Soares Lacerda
Orientador

LAVRAS – MG
2018

RESUMO

O presente trabalho tem como objetivo apresentar o projeto de pesquisa de Mestrado em Engenharia de Sistemas e Automação pela Universidade Federal de Lavras. O projeto proposto consiste em um sistema de identificação de condutores baseado em técnicas de inteligência computacional, por meio de dados de direção do barramento CAN do veículo e sensores inerciais. A investigação se inicia com a avaliação de métodos de seleção e extração de características que otimizem o desempenho pelo algoritmo de aprendizado de máquina *batch*. Estes resultados preliminares tem como objetivo comprovar a viabilidade do sistema. Uma vez terminada esta etapa, será implementado um sistema evolutivo de identificação que foque aperfeiçoamento do algoritmo de classificação a medida que o mesmo for utilizado. Para o desenvolvimento e testes do sistema serão utilizados bancos de dados de direção disponíveis na literatura, que são constituídos de dados do barramento CAN colhidos por meio da interface OBD-II e sensores inerciais presentes em *smartphones*. O sistema implementado será validado em diferentes testes e espera-se atingir uma taxa satisfatória de acertos por parte do sistema na autenticação correta dos condutores avaliados.

Palavras-chave: Identificação de Condutores. Comportamento de Condutores. Aprendizado de Máquina. Processamento de Dados. Aplicações Móveis.

ABSTRACT

The present work has the goal of present the Masters Research Project in Systems and Automation Engineering in the Federal University of Lavras. The proposed project consists of a driver identification system based on machine learning techniques, by means of vehicle CAN bus driving data and inertial sensors. The research begins with the evaluation methods of feature selection and extraction and that optimize the performance of the machine learning algorithm. These preliminary results aim to prove the viability of the system. Once this step is completed, an evolving identification system will be implemented, focusing on the improvement of the classification algorithm as it is used. For the development and testing of the system will be used driving datasets available in the literature, which are constituted of CAN-bus data collected through the OBD-II interface and inertial sensors present in smartphones. The implemented system will be validated in different tests and it will be expect a satisfactory hit rate by the system in the correct authentication of the tested drivers.

Keywords: Driver Identification. Driver Behavior. Machine Learning. Data Processing. Mobile Application.

LISTA DE FIGURAS

Figura 1.1 – Veículos Roubados no Brasil por ano.	7
Figura 2.1 – Exemplos de sistemas de automatização em diferentes níveis.	14
Figura 2.2 – Exemplo de classificação kNN com $k = 5$	25
Figura 2.3 – Representação de uma Floresta Aleatória com n Árvores de Decisão.	26
Figura 2.4 – Representação de um neurônio artificial não linear.	27
Figura 2.5 – Representação de uma MLP com duas camadas escondidas.	28
Figura 2.6 – Representação as diferenças de algoritmos de aprendizado Incremental e Não-Incremental.	29
Figura 3.1 – Visão geral do Sistema de Identificação de Condutores baseado em Técnicas de Inteligência Computacional.	31
Figura 3.2 – Visão geral do Sistema de Identificação de Condutores baseado em Técnicas de Inteligência Computacional.	32
Figura 3.3 – Fluxo de desenvolvimento e testes do sistema preliminar.	37
Figura 4.1 – Precisão média de cada método de redução de dimensionalidade.	39
Figura 4.2 – Precisão média de cada classificador por método de redução de dimensionalidade.	40

LISTA DE TABELAS

Tabela 2.1 – Técnicas de inteligência computacional e suas aplicações em veículos inteligentes.	17
Tabela 2.2 – Interpretação dos valores de Kappa	30
Tabela 3.1 – Características de entrada selecionadas.	35
Tabela 4.1 – Desempenho do Classificador KNN e Técnicas de Redução de Dimensionalidade em diferentes métricas	41
Tabela 4.2 – Desempenho do Classificador <i>Random Forest</i> e Técnicas de Redução de Dimensionalidade em diferentes métricas	41
Tabela 4.3 – Desempenho do Classificador RNA e Técnicas de Redução de Dimensionalidade em diferentes métricas	41
Tabela 6.1 – Equipamentos necessários e orçamento para desenvolvimento do projeto. .	43
Tabela 7.1 – Cronograma das atividades que serão realizadas.	45

SUMÁRIO

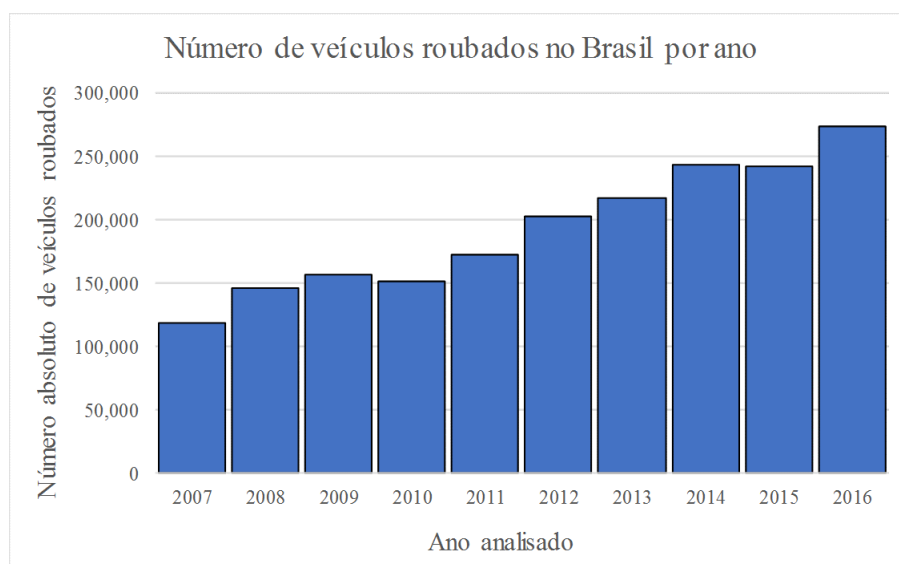
1	INTRODUÇÃO	7
1.1	Objetivos	8
1.2	Motivação	9
1.3	Estrutura do Trabalho	10
2	REVISÃO DA LITERATURA	11
2.1	Trabalhos Correlatos	11
2.1.1	Veículos Inteligentes e Automação Veicular	11
2.1.2	Sistemas Avançados de Assistência ao Condutor (ADAS)	14
2.1.3	Modelagem e Identificação de Comportamento do Condutor	16
2.2	Extração de Características	18
2.2.1	Estatísticas de Ordem Superior	18
2.3	Seleção de Características e Redução de Dimensionalidade	19
2.3.1	Análise de Componentes Principais	20
2.3.2	PCA Incremental	21
2.3.3	Análise Componentes Independentes	21
2.3.4	Análise de Discriminante de Fisher	22
2.4	Algoritmos de Classificação Não-Incrementais (<i>Batch</i>)	23
2.4.1	K-Nearest Neighbors	24
2.4.2	Florestas Aleatórias	25
2.4.3	Redes Neurais Artificiais	26
2.5	Algoritmos de Classificação Incrementais (<i>Online</i>)	28
2.6	Métricas Associadas	29
3	METODOLOGIA	31
3.1	Visão geral do sistema proposto	31
3.2	<i>Datasets</i> Utilizados	32
3.3	Desenvolvimento Preliminar	34
3.3.1	Seleção de Características	34
3.3.2	Extração de Características	35
3.3.3	Redução da Dimensionalidade	35
3.3.4	Classificação	36
3.3.5	Configuração dos experimentos	36

3.4	Modelo Computacional Evolutivo	36
3.4.1	Confuguração de Experimentos Futuras	37
4	RESULTADOS PRELIMINARES E ESPERADOS	38
4.1	Resultados Preliminares	38
4.2	Resultados esperados	41
5	EQUIPE	42
6	ORÇAMENTO	43
7	CRONOGRAMA	44
	REFERÊNCIAS	46

1 INTRODUÇÃO

A cada ano, o número de veículos furtados ou roubados no Brasil tem aumentado consideravelmente. Somente em 2016 foram 557 mil veículos furtados no país, contra 510 mil em 2015, o que representa um aumento de 8% de um ano para o outro, segundo o Fórum Brasileiro de Segurança Pública¹. Outros números desta instituição apontam que 41% destes crimes ocorrem nas capitais. A Figura 1.1 mostra a série histórica de roubos de veículos no Brasil desde 2007, ano em que Fórum Brasileiro de Segurança Pública iniciou a série (os dados referentes a furtos foram contabilizados apenas a partir do ano de 2013) (Fórum Brasileiro de Segurança Pública, 2017).

Figura 1.1 – Veículos Roubados no Brasil por ano.



Fonte: Adaptado de Fórum Brasileiro de Segurança Pública (2017).

Estes números contribuem para a discussão de novas formas de prevenção contra roubos e furtos, de interesse de toda a sociedade, em especial proprietários de veículos, seguradoras de bens e autoridades de segurança pública que diretamente sofrem com as consequências deste problema. Tecnologias que evitam este tipo de situação são vistas com bons olhos, principalmente quando funcionam de forma rápida e de tal modo a deter a ação criminosa a tempo de se evitar perdas maiores.

Grande parte dos dispositivos antifurto veicular em pregados atualmente são restritos a sistemas físicos instalados no veículo, tais como travas elétricas e sistemas de alarme, porém sem nenhum sistema de comunicação externo que alerte o proprietário ou as autoridades de

¹ <https://goo.gl/n6ajJ2>

segurança. Estes sistemas também são facilmente burlados e consequentemente não impedem o criminoso de cometer o delito.

Em paralelo a este contexto, a indústria automobilística tem experimentado avanços no que diz respeito à conectividade, emprego de sensores de diversas finalidades, que viabilizam a introdução do conceito de *smart vehicles*. Isto significa que os veículos estão cada vez mais *cientes* do ambiente no qual estão inseridos e, consequentemente, podem alertar, aconselhar ou até mesmo intervir em situações ditas de risco aos usuários do mesmo. Hoje existem diversos Sistemas Avançados de Assistência ao Condutor, ou ADAS (*Advanced Driver Assistance Systems*) que auxiliam o condutor na dinâmica de direção, além de otimizarem o consumo energético, entre outras funções que serão abordados posteriormente.

Diante do problema brasileiro de furtos e roubos de veículos e ciente do potencial que os *smart vehicles* têm de solucionar problemas relacionados à dinâmica de direção, este trabalho tem como objetivo propor um sistema de identificação de condutores, ou seja um sistema que saiba discernir entre condutores "autorizados" a conduzir o veículo e um possível criminoso que tenha o tenha furtado ou roubado, tomando as devidas providência para impedir a casualidade. Isto se dará pela identificação de certos padrões de direção, característicos de cada condutor, extraídos de dados provenientes de sensores do veículo e externos a este que modelem a dinâmica de direção e posteriormente submetidos à uma técnica de inteligência computacional que identificará o condutor.

1.1 Objetivos

Este trabalho tem como objetivo investigar técnicas de aquisição e processamento de dados, bem como recursos de inteligência computacional e sua aplicabilidade em um sistema de identificação de condutores, com o intuito de identificar possíveis situações de furto ou roubo. Este sistema fará uso de dados oriundos de sensores presentes no veículo e lidos por meio da interface OBD-II e sensores inerciais embarcados em outros dispositivos. Este sistema resultará no desenvolvimento de um modelo computacional que cumprirá a tarefa de identificação do condutor que opera o veículo. Para tal, foram estabelecidos os seguintes objetivos específicos:

- Analisar a viabilidade do sistema proposto por meio do uso de banco de dados utilizando o *dataset* do *Hacking and Countermeasure Research Lab* (KWAK; WOO; KIM, 2016), *dataset* UYANIK (ABUT et al., 2007) e *Driving Behavior Dataset* da PUC-Rio (Oliveira Vasconcelos, 2017).

- Avaliar técnicas de processamento de dados, de modo a eliminar quaisquer ruídos e erros de leitura, e otimizar a extração de características pela técnica de inteligência computacional,
- Após comprovação da viabilidade do sistema, determinar uma técnica de inteligência computacional com aprendizado *online* tendo em vista o melhor desempenho em identificar o condutor. Irá-se avaliar alguns aspectos para a seleção da técnica: 1) a precisão na identificação dos condutores; 2) o custo computacional da mesma, visto que o sistema poderá potencialmente ser embarcado em uma plataforma limitada computacionalmente; 3) O tempo de processamento até que o condutor seja autenticado, uma vez que quando mais rápido um condutor desautorizado for identificado, maiores as chances de evitar uma ação criminosa.
- Modelagem e configuração de experimentos com o sistema de aprendizado online, de forma a simular situações reais e testar o desempenho do modelo de identificação.

1.2 Motivação

Os avanços dos ADAS em países desenvolvidos têm visado atender as necessidades destes países. Ou seja, países com índices de criminalidade baixa, as fabricantes de automóveis não tem como interesse primário desenvolver sistemas anti-furto, tal qual proposto neste trabalho. É preciso identificar as demandas de cada região e então torna-se mais interessante oferecer funcionalidades que proporcionem ao consumidor maior conforto e segurança.

O Brasil, tradicionalmente, é um importador de tecnologias, sendo assim, consomem-se aqui técnicas que não foram desenvolvidas para atender as necessidades do país, surgindo a demanda pela adaptação de tecnologias ADAS à realidade brasileira. Então, ciente desta lacuna que existe por um sistema anti-furto, aliada ao potencial que ADAS tem a oferecer, um sistema de identificação de condutores traria contribuições enumeradas a seguir:

- O sistema de identificação de condutores tem potencial de contribuir na mitigação o problema crônico de roubos e furtos de veículos no Brasil.
- Os prejuízos anuais causados por roubos e furtos de veículos à população, seguradoras e a mobilização de forças de segurança pública na investigação e repressão à este problema, poderiam ser reduzidos pelo emprego de tal sistema.

- Do ponto de vista de investigação de acidentes e responsabilização de casualidades, o sistema poderia ser empregado junto com sistemas caixa-preta na investigação de casualidades e responsabilização dos culpados (BARBOSA, 2017).

1.3 Estrutura do Trabalho

Este trabalho está organizado da seguinte forma: O Capítulo 2 apresenta a revisão bibliográfica acerca dos temas que serão abordados ao longo da pesquisa, tais como veículos inteligentes, sistemas avançados de assistência ao condutor (ADAS) e inteligência computacional aplicada à identificação de comportamento de condutores, além da elucidação acerca das ferramentas e técnicas associadas ao desenvolvimento do modelo. No Capítulo 3, serão apresentados as etapas de desenvolvimento do sistema proposto, bem como uma breve explanação com base na literatura dos procedimentos que serão adotados. No Capítulo 4 serão indicados quais os resultados esperados ao final do projeto. O Capítulo 5 apresenta a equipe envolvida na pesquisa. O orçamento esperado para o projeto está contido no Capítulo 6. Por fim, o Capítulo 7 apresenta o cronograma das atividades a serem desenvolvidas.

2 REVISÃO DA LITERATURA

Este capítulo tem como objetivo apresentar um estudo sobre o estado da arte do sistema proposto bem como as principais ferramentas a serem empregadas no desenvolvimento mesmo.

2.1 Trabalhos Correlatos

Esta seção apresenta o estado da arte relacionada aos veículos inteligentes e aplicações, tratando de tópicos relacionados às normas internacionais que regem o ramo. Outro ponto considerado é a definição de tecnologias assistivas ao condutor (ADAS) e também um estudo bibliográfico sobre as técnicas e abordagens relacionadas à modelagem e identificação de comportamento de condutores.

2.1.1 Veículos Inteligentes e Automação Veicular

A revolução tecnológica que diversos setores têm experimentado se aplica também ao ramo automobilístico. Desde que o primeiro veículo automotor ganhou as ruas no século XIX, tecnologias vem sendo empregadas de forma contínua de tal forma a proporcionar conforto, segurança e economia aos usuários de veículos. Atualmente, a maioria dos veículos em circulação possuem algumas destas tecnologias, como os freios ABS (*Anti-Lock Braking System*) e Programas de Estabilidade Eletrônica (ESP: *Electronic Stability Program*), que de forma contundente proporcionam segurança aos condutores e passageiros por meio do uso de sistemas eletrônicos de alta qualidade.

Esta evolução tecnológica se deve muito à popularização de dispositivos semicondutores que deu início à Era dos Computadores na década de 1970, onde, visto o que o uso de sistemas microprocessados poderiam proporcionar, já se começou a imaginar um futuro onde veículos completamente autônomos estariam inseridos no mercado.

Seguindo a evolução e popularização dos microprocessadores, surgiram assim as Unidades de Controle Eletrônico (ECU). Na indústria automotiva, uma ECU é um dispositivo eletrônico embarcado que realiza a leitura de sinais oriundos de sensores localizados em diversas partes e componentes do veículo e dependendo destas informações colhidas controla várias partes importantes do veículo, como o motor e outras opções automatizadas do veículo (EBERT; JONES, 2009).

A grande quantidade de informações que passou a circular entre ECU's fez com que surgisse a necessidade de desenvolver uma rede de comunicação multiplexada entre as ECU's, de tal forma a simplificar o cabeamento e reduzir custos de implementação. Assim na década de 80, foi desenvolvido pela Bosch a *Controller Area Network*, ou CAN-bus, que é um barramento intraveicular de comunicação entre ECU's, sensores e atuadores. O CAN-bus é conhecido por sua robustez na transmissão de dados, sendo resistente à interferências eletromagnéticas, e operação em tempo real (TUOHY et al., 2015). Sendo assim, a crescente capacidade de processamento das ECU's, aliado à grande quantidade de informações relevantes disponíveis no CAN-bus, proporcionaram a introdução do conceito de veículos inteligentes.

Veículos inteligentes, segundo Hubaux, Čapkun e Luo (2004), são veículos que são capazes de perceber o ambiente no qual estão inseridos. Um veículo inteligente deve estar equipado com registradores de dados, processadores, sistemas de posicionamento e geolocalização, sensores que permitam que o veículo esteja ciente do ambiente, seja no contexto intraveicular, ou no contexto extraveicular.

No contexto intraveicular, o veículo está ciente de diversos fatores no que diz respeito ao comportamento do condutor, tais como a percepção de fatores neurofisiológicos que podem comprometer a segurança dos usuários, como detecção de sonolência, agressividade ou embriaguez, através de padrões de direção característicos de cada situação.

No que diz respeito ao contexto extraveicular, o veículo percebe uma série de situações que podem comprometer a segurança dos usuários e, conseqüentemente, alertar ou até mesmo intervir na dinâmica de direção de modo evitar casualidades. Como exemplos pode-se citar o monitoramento de pontos cegos do veículo, como uma forma de proporcionar ultrapassagens seguras. Sistemas de estacionamento automatizado são outro exemplo deste tipo de sistema, onde o veículo monitora os objetos que o rodeiam e realiza a manobra de estacionamento de forma automática. Outro sistema que visa garantir a segurança dos usuários é o controle de velocidade de cruzeiro, onde o veículo realiza o ajuste da velocidade de acordo com a distância entre o veículo hospedeiro e os veículos à frente. Este sistema é baseado na velocidade do veículo à frente o veículo hospedeiro desacelera ou acelera até a velocidade limiar, tal que a distância entre os veículos seja a estipulada para a segurança dos usuários.

A medida que os veículos inteligentes passam a ser mais sensíveis ao ambiente, mais próximos à total automatização da direção eles estão. Segundo De Winter et al. (2014), diversos grupos de pesquisa ao redor mundo focam no estudo de veículos autônomos com o objetivo de

uma inserção revolucionária do produto no mercado, porém, na prática, este processo é mais evolucionário que revolucionário. Isto se dá pelo fato que ano após ano, novas tecnologias de assistência ao condutor (assunto a ser tratado na próxima seção) são inseridos no mercado, e assim mais automatizada se torna a direção.

Segundo a norma J3016 da **SAE INTERNATIONAL (2014)**, existem seis níveis de automação veicular, desde não automatizado até totalmente automatizado. Dos níveis 0 ao 2 é esperado que o *Condutor Humano* monitore o ambiente de direção e que seja responsável pelas decisões tomadas, porém podem existir sistemas que auxiliem o condutor na tomada de decisões. Por sua vez, nos níveis de 3 a 5, o responsável pelo monitoramento de direção é o *Sistema de Direção Automático*, que pode ser definido como a combinação de diversos sistemas de assistência ao condutor (DAS), porém dos quais não estão inclusos sistemas de intervenção e advertência momentânea pelo fato de não automatizarem partes da tarefa dinâmica de direção e, consequentemente, não mudarem o papel do condutor humano. Os seis níveis de automatização da direção são:

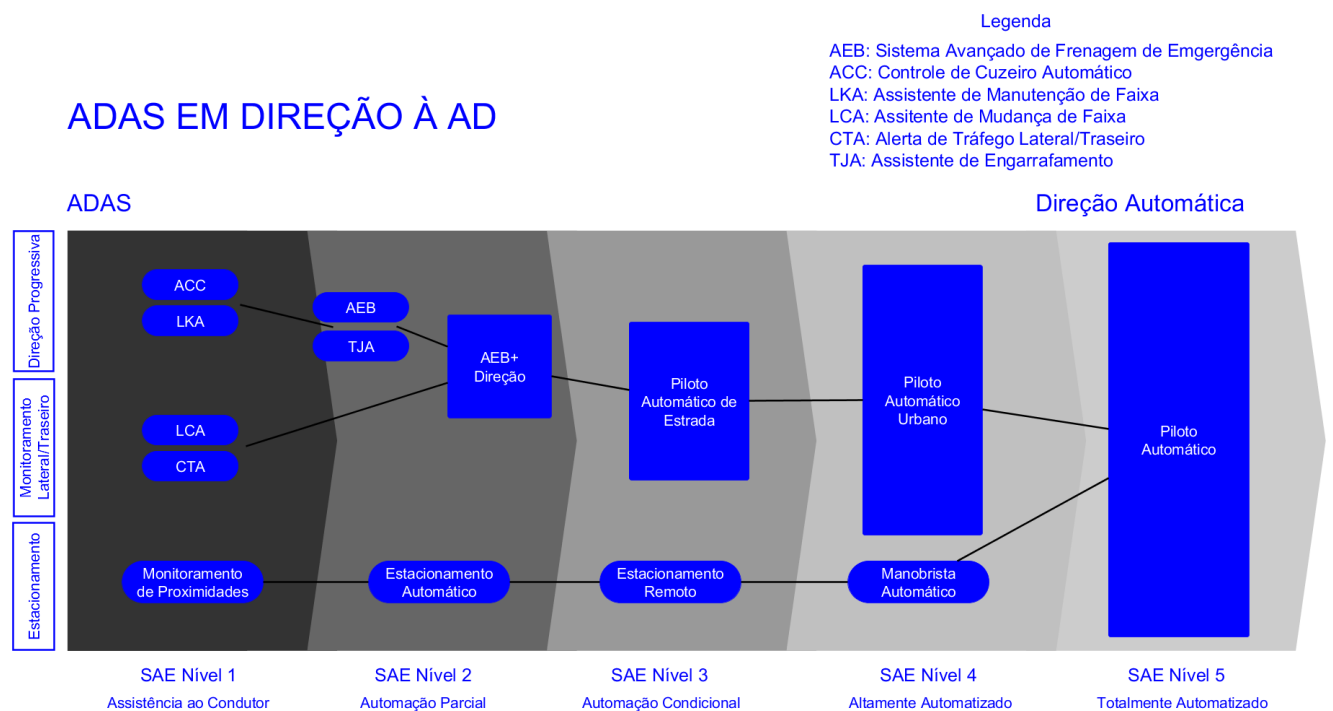
- **SAE Nível 0 - Não Automatizado:** O condutor humano controla todos os aspectos da tarefa dinâmica de condução, mesmo quando auxiliado por sistemas de alerta e intervenção. O condutor humano é responsável pelo monitoramento do ambiente.
- **SAE Nível 1 - Assistência ao Condutor:** O modo de direção-execução específica é feito pelo sistema ou pelo condutor, tanto na direção, quanto na aceleração/desaceleração, usando informações do ambiente enquanto o condutor humano é responsável por todas os outros aspectos da dinâmica de condução.
- **SAE Nível 2 - Automação Parcial:** O modo de direção-execução específica é feito por um ou mais sistemas de assistência, tanto na direção, quanto na aceleração/desaceleração, usando informações do ambiente enquanto o condutor humano é responsável por todas os outros aspectos da dinâmica de condução.
- **SAE Nível 3 - Automação Condicional:** O modo de direção específica é feito por um sistema automatizado de condução em todos os aspectos da tarefa dinâmica de condução com a expectativa de que o condutor humano assuma o controle se requisitado.
- **SAE Nível 4 - Altamente Automatizado:** O modo de direção específica é feito por um sistema automatizado de condução em todos os aspectos da tarefa dinâmica de condução,

mesmo que o condutor humano não assuma o controle apropriadamente quando requisitado.

- **SAE Nível 5 - Totalmente Automatizado:** O modo de direção específica é feito por um sistema automatizado de condução em todos os aspectos da tarefa dinâmica de condução sob quaisquer condições ambientais.

Como exemplificação destes níveis de automatização, a Figura 2.1 ilustra os tipos de tecnologias em cada nível SAE, desde somente sistemas avançados de assistência ao condutor, até sistemas de direção automatizados.

Figura 2.1 – Exemplos de sistemas de automatização em diferentes níveis.



Fonte: Adaptado de Renesas (2017)

2.1.2 Sistemas Avançados de Assistência ao Condutor (ADAS)

Acidentes de trânsito são uma das maiores causas de mortes na atualidade. A maioria dos acidentes de trânsito são provocados pela desatenção e imprudência dos condutores. Segundo Hoeger et al. (2009), 97% dos acidentes são causados por falha humana. Contudo, grande parte dos estudos são voltados para macro-problemas, como o número de veículos nas vias e gerenciamento de tráfego, e não dão atenção suficiente para decisões individuais de condutores e como isso impacta no tráfego (CARMONA et al., 2015). Portanto, assistência ao

condutor em diferentes níveis são ferramentas potenciais para a segurança dos usuários de veículos e pedestres. Ciente disso, os recentes avanços em inteligência computacional e técnicas de percepção têm proporcionado uma série de novas aplicações desenvolvidas com o propósito de prevenir estes tipos de casualidades utilizando o conceito chamado de Sistemas Avançados de Assistência ao Condutor, ou simplesmente ADAS (*Advanced Driver Assistace System*).

Segundo Saito, Itoh e Inagaki (2016) os ADAS são baseados em modelos de automação e podem ser categorizados em quatro classes:

1. Percepção Aumentada: São todos os sistemas que têm por finalidade monitorar situações em que normalmente o condutor não tem condições de controlar com precisão. Como exemplo estão sensores de estacionamento, monitoramento de ponto cego em mudança de faixa ou ultrapassagem.
2. Aleta para riscos potenciais: Sistemas que têm intuito de advertir o condutor sobre alguma situação de risco, tais como sonolência/fadiga, pouca distância ao veículo à frente, alerta sobre possíveis falhas nos sistemas do veículo se enquadram nesta classe.
3. Avisos desencadeados: São sistemas que solicitam ao condutor que tome uma ação específica em uma determinada situação de risco. Como exemplos estão alertas de velocidade alta e detecção de sonolência.
4. Controle de segurança automático: Este sistema é acionado quando o condutor não toma providências quando alertado ou a ação de controle do condutor é insuficiente para tal situação de risco.

As classes 1 e 2 são implementadas para auxiliar o condutor a perceber ou entender o contexto. Este entendimento determina quais ações devem ser tomadas. Uma vez que decisão de diagnóstico de situação é feita, a seleção de ações efetuadas pelo condutor geralmente é a adequada. Contudo, a decisão tomada pelo condutor pode não ser a correta. Assim a classe 3 auxilia o condutor em tal circunstância. Qualquer ADAS que utilize as classes 1, 2 e 3 são compatíveis com o princípio de automação centrada no ser humano, onde o condutor é a autoridade final no processo de automação. Se o ADAS conter a classe 4, a autoridade passa a ser dividida entre o sistema e o humano. Existem controvérsias sobre o uso da classe 4, uma vez que máquinas altamente automatizadas podem trazer efeitos negativos, como falhas em sensores, erros na malha de controle, complacência, excesso de confiança no sistema, entre outros (INAGAKI, 2012).

2.1.3 Modelagem e Identificação de Comportamento do Condutor

A revisão bibliográfica feita por Meiring e Myburgh (2015) aborda quais algoritmos de inteligência artificial são mais adequados para análise de estilos de direção e comportamento de condutores. É abordado quais os tipos de direção, bem como as causas e consequências de cada um deles (normal/seguro, agressivo, desatento (fadiga, distração), alcoolizado), e aponta o potencial de tais algoritmos em detectar tais condições e prevenir casualidades proporcionadas pelas mesmas. É também realizada uma classificação de técnicas de inteligência computacional e suas aplicações mais comuns são apresentadas na Tabela 2.1. Técnicas de inteligência computacional e suas aplicações em ADAS. É concluído que os algoritmos mais promissores para criação de aplicações futuras em ADAS são técnicas baseadas em Lógica Fuzzy, implementação de Modelos Ocultos de Markov (HMM) e Máquinas de Vetor de Suporte (SVM).

Em seu trabalho, Kumtepe, Akar e Yuncu (2016) propõem um modelo que faz a fusão de dados provenientes de sensores disponíveis no carro e através de câmeras com o objetivo de se decidir se o condutor apresenta sinais de agressividade ou desatenção. Estas informações são usadas para formar o vetor de características que representam o comportamento do condutor e então são submetidos a uma máquina de vetores de suporte (SVM) de modo a se classificar o condutor testado. O método proposto obteve uma taxa de detecção de agressividade por parte do condutor de 93.1%.

Por sua vez, Blaszczyk, Turek e Cetnarowicz (2014) realiza experimentos controlados, onde dois condutores são submetidos a um teste em circuito fechado e dados são coletados por meio da interface OBD-II e Raspberry Pi, além do uso de sensores embarcados em um *smartphone*, com o objetivo de diferencia-los. Utilizando quatro técnicas de classificação (Processo Gaussiano, M5P, *M5Rules* e Tabela de Decisões). O erro médio quadrático (RMSE) para Processo Gaussiano, M5P, *M5Rules* e Tabela de Decisões, foi, respectivamente 0.168, 0.087, 0.023 e 0. Apesar de o erro relativamente baixo, é importante ressaltar que somente dois condutores foram comparados, necessitando assim de mais testes de modo a comprovar a eficácia do sistema.

Seguindo outra vertente, Castignani et al. (2015) faz uso somente de *smartphones* para monitorar o comportamento do condutor e alertar para possíveis situações de risco. O uso do dispositivo se justifica pela sua atual penetração no mercado e a facilidade de desenvolvimento de aplicativos, além da robustez dos sensores embarcados nos aparelhos. Um sistema fuzzy é usado para computar o escore de diferentes condutores usando informações em tempo real,

Tabela 2.1 – Técnicas de inteligência computacional e suas aplicações em veículos inteligentes.

Técnica	Aplicações
Redes Neurais Artificiais	Detecção de sonolência e distração, predição do comportamento do volante, visão computacional.
Fast Fourier Transform	Detecção de sonolência.
Clusterização	Distinção de estilo de direção e rotulação de condutor.
Clusterização K-means	Identificação individual de condutor e monitoramento de condições de rota.
Máquina de estados	Reconhecimento de manobras.
Máquina de estados finitos	Modelagem de tomadas de decisão do condutor.
Máquina de estados híbridos	Veículos Autônomos.
Lógica Fuzzy	Detecção de fadiga, identificação de distração, métodos de pontuação e métodos de reconhecimento de estilo de direção.
Modelos Ocultos de Markov (HMM)	Estimação de comportamento do condutor, reconhecimento de manobras, análise de performance de direção e identificação de distração.
Técnicas Bayesianas	Estimação de comportamento do condutor em situações de dados faltantes.
Árvores de decisão	Estimação de confiança de resultados em fusão de dados para detecção de sonolência.
Modelo de misturas de gaussianas	Identificação de distração, reconhecimento de manobras e monitoramento de condições de rota.
Dynamic time warping (DTW)	Classificação de perfil de risco do condutor e assistentes de direção ou alerta de segurança.
Filtros de Kalman	Predição de processos e modelagem de comportamento humano.
Máquinas de vetor de suporte (SVM)	Métodos de reconhecimento de estilo de direção, detecção de sonolência e estimação de estado do veículo.
Algoritmos Genéticos	Calibração de processos de modelagem de carros adjacentes.

Fonte: Adaptado de Meiring e Myburgh (2015).

como acelerômetros e sistema de navegação GPS provenientes do *smartphone*, aceleração, frenagens e posição do volante, provenientes do barramento CAN do veículo por meio da interface OBD-II, além de topologia da rota e condições climáticas. A validação do sistema é feita através de teste em circuito fechado com diversos condutores.

Técnicas de inteligência computacional híbridas também são exploradas, como no trabalho de Echanobe, Campo e Martinez (2016), onde uma Rede Neural Artificial (RNA) é otimizado por Algoritmos Genéticos Multiobjetivos com a finalidade de realizar a classificação de condutores através de diversos parâmetros de entrada. Testes foram conduzidos com diferentes parâmetros e número de neurônios e camadas, de modo a diminuir o erro da RNA. O resultado

obtido é um classificador com menor número de variáveis de entrada em diferentes domínios (tempo, frequência e cepstral), menor quantidade de neurônios e camadas escondidas da RNA e por consequência um sistema mais rápido eficiente no que diz respeito ao custo computacional.

2.2 Extração de Características

Em aprendizado de máquina, reconhecimento de padrões e processamento de imagem, extração de características começa por um conjunto inicial de dados medidos e retorna valores derivados (características) com a intenção de serem informativas e não redundantes. A aplicação destas técnicas facilitam os subsequentes passos de aprendizagem e generalização levando a uma melhor interpretação dos dados.

2.2.1 Estatísticas de Ordem Superior

O termo Estatística de Ordem Superior (EOS) refere à funções que usam a terceira potência ou superiores de uma certa amostra, diferentemente de Estatística de Ordem Inferior, que fazem uso de termos constantes, lineares e quadráticos (potências zero, um e dois). EOS pode ser definidos em termos de momentos e cumulantes. enquanto momentos são próprios para descrever sinais determinísticos, cumulantes são adequados para a análise de sinais estocásticos (dados de direção podem ser classificados como estocásticos) (GUEDES; FERREIRA; BARBOSA, 2016):. Considere x como um processo aleatório, real e discreto com média zero. Então, a segundo, terceiro e quarto cumulante podem ser obtidos, respectivamente por

Dado um caso unidimensional ($d = 1$), os momentos de uma variável aleatória X podem ser definidos como:

$$\begin{aligned} m_1 &= \langle x \rangle \\ m_2 &= \langle x^2 \rangle \\ &\vdots \\ m_n &= \langle x^n \rangle \end{aligned} \tag{2.1}$$

Então os cumulantes podem ser escritos em forma de momentos:

$$\begin{aligned}
c_1 &= m_1 \\
c_2 &= m_2 - m_1^2 = \sigma^2 \\
c_3 &= m_3 - 3m_1m_2 + 2m_1^3 \\
c_4 &= m_4 - 3m_2^2 - 4m_1m_3 + 12m_1^2m_2 - 6m_1^4
\end{aligned} \tag{2.2}$$

Os cumulantes c_1 , c_2 , c_3 e c_4 são a média, a variância, a assimetria e a curtose, respectivamente. Dentre os quais somente a média não é uma EOS. Para um sinal discreto $x[n]$, os cumulantes podem ser obtidos de forma direta por:

$$C_{2,x}(\tau) = \frac{1}{N} \sum_{n=0}^{N-1} x[n]x[\text{mod}(n + \tau, N)], \tag{2.3}$$

$$C_{3,x}(\tau) = \frac{1}{N} \sum_{n=0}^{N-1} x[n]x^2[\text{mod}(n + \tau, N)], \tag{2.4}$$

$$\begin{aligned}
C_{4,x}(\tau) &= \frac{1}{N} \sum_{n=0}^{N-1} x[n]x^3[\text{mod}(n + \tau, N)] - \\
&\quad 3 \frac{1}{N^2} \sum_{n=0}^{N-1} x[n]x[\text{mod}(n - \tau, N)] \cdot \sum_{n=0}^{N-1} x^2[n]
\end{aligned} \tag{2.5}$$

onde $x \in \Re^N$, $\tau = [0, 1, \dots, N-1]$ são os atrasos, e *mod* é o operador módulo, que retorna o resto de uma operação inteira (MOREIRA; FERREIRA; DUQUE, 2014).

Como descrito em Mendel (1991), EOS pode levar a resultados mais representativos quando aplicadas como ferramenta de extração de características em processos não lineares e não gaussianos. De acordo com Naves, Barbosa e Ferreira (2016), a maior vantagem do uso dos cumulantes como extrator de características em problemas de classificação é sua propriedade de imunidade ao ruído gaussiano.

2.3 Seleção de Características e Redução de Dimensionalidade

Uma ferramenta importante para melhorar a eficiência da tarefa de classificação em termos de custo computacional é a redução de dimensionalidade. Esta ferramenta pode ser definida como o processo de redução do número de variáveis de entrada pela obtenção de um conjunto de variáveis principais. Em outros termos, estas técnicas transformam um conjunto de dados de amplo espaço dimensional em um espaço de menor dimensão.

A técnica mais utilizada para redução de dimensionalidade é a Análise de Componentes Principais, descrita posteriormente neste trabalho. Outra técnica que pode ser explorada para esta tarefa é a Análise de Componentes Independentes, embora esta não seja convencionalmente usada para redução de dimensionalidade, mas para a separação de sinais sobrepostos. Contudo, existem exemplos na literatura que aplicam ICA para redução de dimensionalidade (WANG; CHANG, 2006) (CAO et al., 2003).

2.3.1 Análise de Componentes Principais

PCA é um procedimento matemático que usa uma transformação ortogonal para converter um conjunto de variáveis possivelmente correlatadas em um conjunto de variáveis não correlacionadas chamadas de componentes principais (WOLD; ESBENSEN; GELADI, 1987). Sejam $x_t (t = 1, \dots, l$ e $\sum_{t=1}^l x_t = 0$) um conjunto de vetores de entrada de m dimensões $x_t = (x_t(1), x_t(2), \dots, x_t(m))^T$, então PCA transforma linearmente x_t em um novo vetor s_t pela seguinte expressão:

$$s_t = U^T x_t, \quad (2.6)$$

onde U é uma matriz ortogonal de $m \times m$ dimensões no qual a i -ésima coluna u_i é o i -ésimo autovetor da matriz de covariância C . Portanto, PCA primeiro soluciona o problema dos autovalores:

$$\lambda_i u_i = C u_i, \quad i = 1, \dots, m, \quad (2.7)$$

onde λ_i é o i -ésimo autovalor de C e u_i é seu o autovetor relativo. Então, os componentes de s_t , baseado no obtido em u_i , são calculados como a transformação ortogonal de x_t por:

$$s_t(i) = u_i^T x_t, \quad i = 1, \dots, m. \quad (2.8)$$

Estes novos componente $s_t(i)$ são os componentes principais. Tal qual a Equação 2.8 demonstra, o número de componentes principais de s_t pode ser reduzido pelo uso de somente alguns dos primeiros autovetores ordenados na ordem descendente dos autovalores. Assim, pode-se concluir que PCA tem a característica de reduzir a dimensão de um conjunto de dados (CAO et al., 2003).

2.3.2 PCA Incremental

O PCA é uma ferramenta útil para problemas para redução de dimensionalidade, porém possui certas limitações quando utilizados em datasets maiores. Isto se dá porque o processamento do PCA é feito em lotes, que faz com que todos os dados a serem processados devam ser armazenados na memória do dispositivo utilizado. Isto pode ser um problema para aplicações em *hardware* embarcado que tem limitações quanto a capacidade de memória (PEDREGOSA et al., 2011).

Diante deste problema, a técnica de PCA Incremental (IPCA) contorna este problema, utilizando uma forma diferente de processamento que permite cálculos parciais que praticamente obtém o mesmo resultado do PCA, porém, realizando o processamento em pequenos lotes (WENG; ZHANG; HWANG, 2003).

A versão incremental do algoritmo funciona da seguinte maneira: assume-se que já se possui o conjunto de autovetores $U = [u_j], j = 1, \dots, p$ do vetor de entrada $x_i, i = 1, \dots, n$. os autovalores correspondentes são $\lambda = \text{diag}(\Lambda)$ e a média dos valores é \bar{x} . A construção dos incrementos requer a atualização destes autovalores e autovetores levando em conta uma nova entrada x_{n+1} . Primeiramente é feita atualização da média por meio da seguinte expressão (ARTAC; JOGAN; LEONARDIS, 2002):

$$\bar{x}' = \frac{1}{n+1} (n\bar{x} + x_{n+1}) \quad (2.9)$$

A atualização dos autovetores é feita pela adição do novo vetor e a aplicação de uma transformação rotacional. Para tal, calcula-se o vetor ortogonal residual $h_{n+1} = (Ua_{n+1} + \bar{x}) - x_{n+1}$ e em seguida sua normalização \hat{h}'_{n+1} . A nova matriz U' é calculada por:

$$U' = [U\hat{h}'_{n+1}]R, \quad (2.10)$$

onde R é a matriz de rotação. Este processo é repetido de acordo com o tamanho de lotes definidos (*batch size*). Isto resulta em um processamento mais eficiente em termos de utilização de memória (BALSUBRAMANI; DASGUPTA; FREUND, 2013).

2.3.3 Análise Componentes Independentes

Análise Componentes Independentes (ICA) (LEE, 1998) é uma técnica que originalmente foi desenvolvida para separação cega de fontes, que recupera sinais mutualmente in-

dependentes, mas com fontes desconhecidas a partir de suas misturas lineares sem saber os coeficientes de mistura.

ICA considera que os dados são linearmente combinados por um conjunto de fontes independentes e estes sinais podem ser separados de acordo com sua independência estatística (WANG; CHANG, 2006). Seja x_t a mistura linear e s_t denota o sinal original, então o objetivo do ICA é estimar s_t por:

$$s_t = Ux_t, \quad (2.11)$$

onde U é a matriz $m \times m$ de separação de misturas. Os componentes retornados por s_t são tão independentes estatisticamente quanto possível.

Existem um amplo número de algoritmos que foram desenvolvidos para executar o ICA. Neste trabalho, considerou-se o FastICA de ponto fixo, proposto por Hyvarinen (1999). Este algoritmo é considerado um dos melhores e mais usados métodos já desenvolvidos. FastICA faz uso de informações mútuas como critério para estimar s_t , ao passo que também é uma medida natural de independência entre variáveis aleatórias. A maximização da negentropia (medida do grau de organização do sistema) corresponde à minimização das informações mútuas entre os componentes. Entretanto, esta negentropia não pode ser feita diretamente uma vez que as densidades de probabilidade dos componentes são desconhecidos. A explanação completa do funcionamento do algoritmo FastICA é encontrada em (KOLDOVSKY; TICHAUSKY; OJA, 2006).

As duas principais diferenças entre o PCA e o ICA são, primeiramente, os componentes retornados pelo ICA são estatisticamente independentes, não simplesmente decorrelacionadas tal qual ocorre nos gerados pelo PCA. A segunda distinção é que a matriz de separação de misturas do ICA não é ortogonal tal qual a do PCA (CAO et al., 2003).

2.3.4 Análise de Discriminante de Fisher

Análise de Discriminante de Fisher (FDA) é uma técnica de redução de dimensionalidade, otimizado em termos da maximização da separação entre classes. Dado um conjunto de dados de $n \times m$ dimensões representado pela matriz X com vetor coluna x_i , a matriz de dispersão total é dada por (CHIANG; KOTANCHEK; KORDON, 2004):

$$S_t = \sum_{i=1}^n (x_i - \mu)(x_i - \mu)^T \quad (2.12)$$

onde μ é o vetor de médias totais dos elementos correspondentes às colunas de X . Considerando X_j como o conjunto de vetores x_i que pertencem à uma determinada classe j , a matriz de dispersão interna para a classe j é

$$S_t = \sum_{x_i \in X_j}^n (x_i - \mu_j)(x_i - \mu_j)^T \quad (2.13)$$

onde μ_j é o vetor de médias para a classe j . Considerando c o número de classes dos dados, então

$$S_w = \sum_{i=1}^c S_i \quad (2.14)$$

é a matriz de dispersão entre classes, onde n_j é o número de observações na classe j .

O primeiro vetor FDA w_1 pode ser determinado como:

$$\operatorname{argmax}_{w_1} \frac{w_1^T S_b w_1}{w_1^T S_w w_1} \quad (2.15)$$

O segundo vetor FDA é calculado de modo a maximizar a dispersão entre classes enquanto minimiza a dispersão dentro das classes entre todos os eixos perpendiculares com o primeiro vetor FDA em seguida com os vetores FDA restantes. Pode ser provado matematicamente que os vetores FDA são iguais aos autovetores w_k do problema de autovalores generalizados

$$S_b w_k = \lambda_k S_w w_k, \quad (2.16)$$

onde os autovalores λ_k indicam o grau geral de separabilidade entre as classes pela projeção de todas as classes em w_k . Com os vetores FDA calculados, as observações são então classificadas de forma a reduzir o espaço dos vetores FDA através de uma análise discriminante (SUGIYAMA, 2007).

2.4 Algoritmos de Classificação Não-Incrementais (*Batch*)

Em Aprendizado de Máquina, mais especificamente em Aprendizado Supervisionado, classificação é o problema de identificar em qual classe um conjunto de observações pertence,

com base em um conjunto de dados de treinamento o qual as classes são conhecidas. Identificação de condutores se enquadra neste problema, uma vez que o objetivo do sistema a ser desenvolvido pelo presente trabalho é reconhecer se um condutor é ou não autorizado a conduzir o veículo. De tal modo, esta seção tem o objetivo apresentar algumas técnicas de Aprendizado de Máquina para problemas de classificação.

2.4.1 K-Nearest Neighbors

k-Nearest Neighbors (kNN) é uma das mais simples e fundamentais técnicas de classificação, adequada principalmente para casos onde não se tem conhecimento *a priori* sobre a distribuição dos dados utilizados. De acordo com Peterson (2009), a classificação kNN foi desenvolvida pela necessidade de realizar análise discriminante quando estimativas paramétricas confiáveis de densidades de probabilidade são desconhecidos ou de difícil determinação. A ideia principal do kNN é determinar a classe de uma amostra, baseado nas amostras vizinhas provenientes de um conjunto de treinamento.

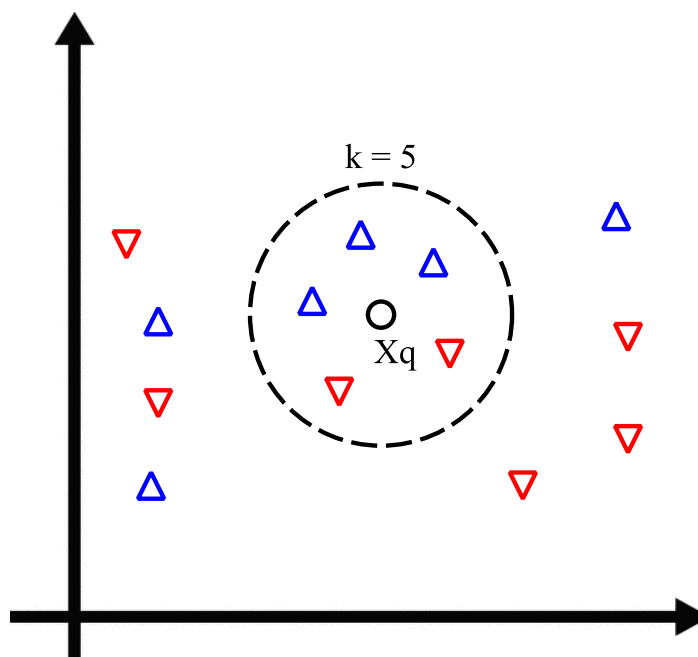
O classificador kNN é predominantemente baseado na distância Euclidiana entre as amostras de teste e a amostra de treino especificada. Considerando que \mathbf{x}_i são as amostras de entrada com p atributos $(x_{i1}, x_{i2}, \dots, x_{ip})$, n é a quantidade total de amostras de entrada ($i = 1, 2, \dots, n$) e p a quantidade total de atributos ($j = 1, 2, \dots, p$). A distância Euclidiana d entre as amostras \mathbf{x}_i e \mathbf{x}_l ($l = 1, 2, \dots, n$) é definida por:

$$d(\mathbf{x}_i, \mathbf{x}_l) = \sqrt{(x_{i1} - x_{l1})^2 + (x_{i2} - x_{l2})^2 + \dots + (x_{ip} - x_{lp})^2} \quad (2.17)$$

Esta distância d é utilizada determinar a classe de treinamento que um determinado elemento desconhecido pertença, o classificador kNN realiza a busca por k elementos do conjunto de treinamento que estejam mais próximos deste elemento desconhecido, isto é, elementos com maior proximidade a este. O kNN é um classificador que possui somente um parâmetro de entrada (k -vizinhos) que normalmente é um número ímpar, para que se evite empates entre as classes e obtido empiricamente ou com o auxílio de técnicas de otimização.

A Figura 2.2 ilustra um exemplo simples de classificação utilizando o kNN, onde um elemento desconhecido x_q é inserido no espaço amostral de classes conhecidas. São obtidos os cinco elementos mais próximos à x_q e é determinada a classe deste a partir da classe com mais elementos dentro deste espaço de busca.

Figura 2.2 – Exemplo de classificação kNN com $k = 5$.



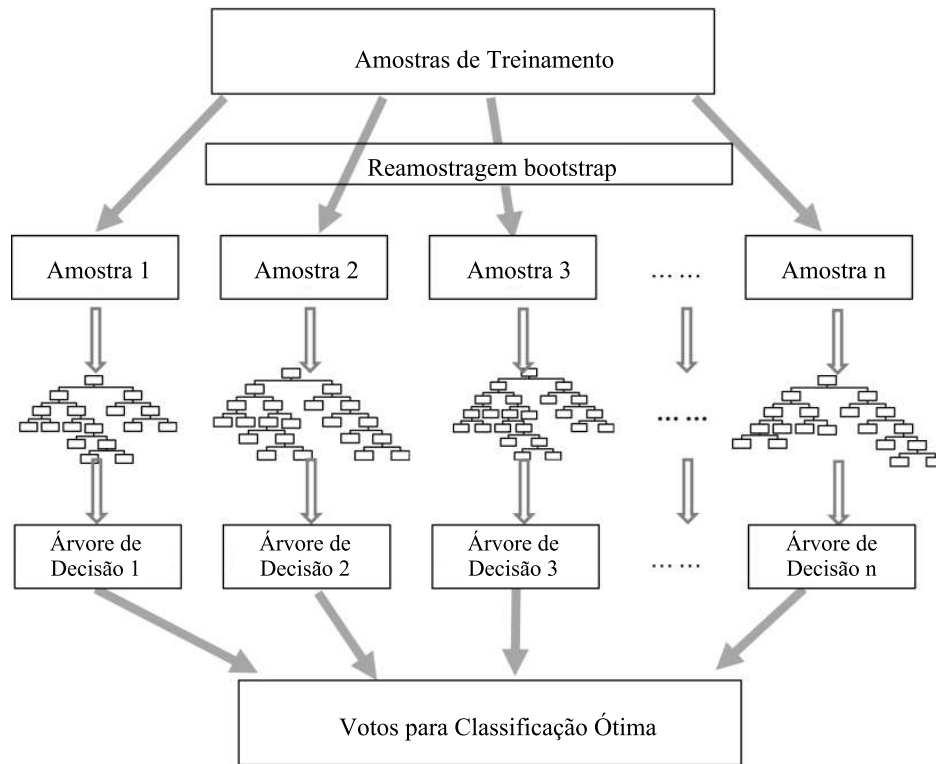
Fonte: Elaborado pelo Autor (2018).

2.4.2 Florestas Aleatórias

Florestas Aleatórias (tradução literal de *Random Forest*) é uma técnica de inteligência computacional para classificação e regressão introduzido por Breiman (2001) que consiste no agrupamento de diversas árvores de decisão (SAFAVIAN; LANDGREBE, 1991) de maneira que sua estrutura seja composta de forma aleatória (ver Figura 2.3). Em árvores de decisão comuns, cada nó é dividido de modo a se obter a melhor divisão entre as variáveis do problema em questão. Por sua vez, florestas aleatórias cada nó é dividido usando o melhor entre os subconjuntos de preditores escolhidos aleatoriamente no nó em questão. O classificador requer somente a configuração de dois parâmetros de entrada para a geração do modelo de predição: o número de árvores de decisão desejadas (n_{tree}) e o número de variáveis de predição (m_{try}) usados em cada nó para o crescimento da árvore.

Esta estratégia produz resultados satisfatórios se comparados a outros classificadores, incluindo análise discriminante, máquinas de vetor de suporte e redes neurais artificiais, além de ser robusto contra o problema de *overfitting* (ou sobre-ajuste: quando um modelo se ajusta satisfatoriamente ao conjunto de dados treinado, mas se mostra ineficaz para prever novos resultados) (BREIMAN, 1999) e boa imunidade à ruído gaussiano (ZHANG et al., 2017). Porém, apesar desta imunidade com relação ao *overfitting* as florestas aleatórias tendem a saturar do

Figura 2.3 – Representação de uma Floresta Aleatória com n Árvores de Decisão.



Fonte: Adaptado de Zhang et al. (2017).

erro final, isto é, o erro se estagna em um ponto mesmo com a inserção de novas árvores à floresta.

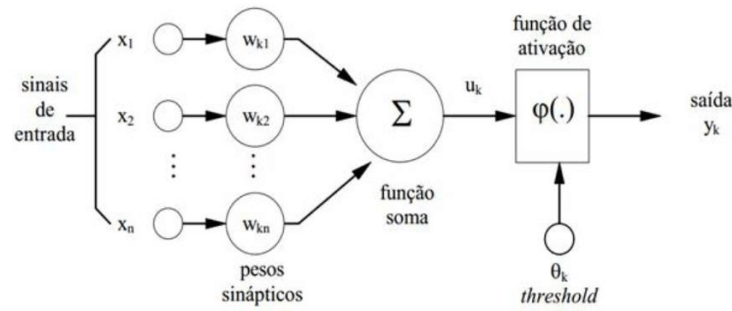
2.4.3 Redes Neurais Artificiais

Um das mais difundidas técnicas de Aprendizado de Máquina, as Redes Neurais Artificiais (RNAs) são amplamente empregadas na solução de diversos problemas de regressão e classificação, atribuindo seu sucesso à sua flexibilidade de síntese de mapeamento multidimensional não-linear de variáveis dependentes e independentes, devido a sua capacidade de aproximação universal.

RNAs são formadas por conjuntos de diversos neurônios artificiais, estes propostos por McCulloch e Pitts em 1943. Este modelo de neurônio consiste em receber sinais de entrada x_n nos dendritos do neurônio e retorna um único sinal de saída y no axônio do mesmo. Este modelo é apresentado na Figura 2.4.

A expressão que representa este neurônio, com função de ativação unitária, é apresentado pela Equação 2.18,

Figura 2.4 – Representação de um neurônio artificial não linear.



Fonte: Adaptado de Haykin et al. (2009).

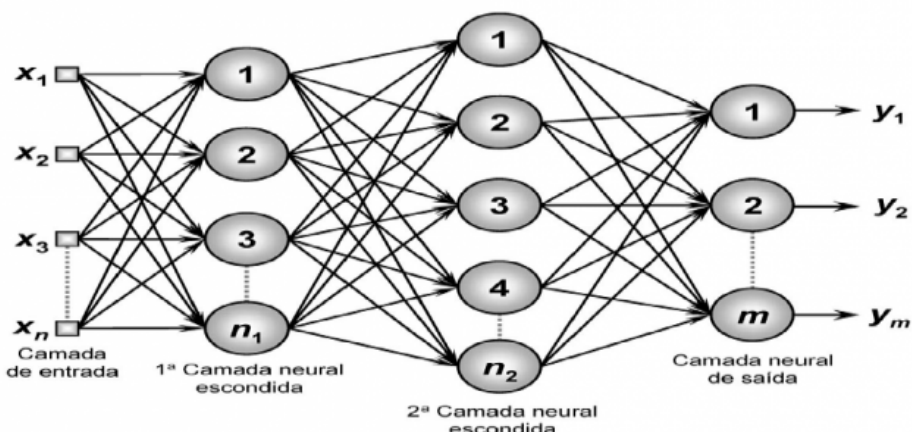
$$u_k = y_k = \sum_{i=1}^m w_i x_i + b \quad (2.18)$$

onde x_i são as entradas dos neurônios, w_i são os pesos das entradas, b é o *bias* e y é a saída para uma função de ativação unitária. A função de ativação é o elemento que determina a saída do neurônio em termos do potencial de ativação, muitas das vezes limitando a saída entre $\{0, 1\}$ e $\{-1, 1\}$. Dentre as funções mais utilizadas para esta operação estão as funções sigmoideal, linear, tangente hiperbólica, logarítmica e senoidal.

As combinações destes neurônios formam uma Rede Neural Artificial. Dentre as muitas arquiteturas utilizadas, a mais comum é a rede com múltiplas camadas. Haykin et al. (2009) definiu estas redes como sendo um conjunto de unidades sensoriais que constituem a camada de entrada, uma ou mais camadas ocultas e uma camada de saída. Estas redes são conhecidas como Perceptron Multicamadas (*Multilayer Perceptron* - MLP) (ROSENBLATT, 1962). Na Figura 2.5, é apresentado um modelo de uma MLP com duas camadas escondidas (*hidden layers*).

Existem diversas estratégias de treinamento supervisionado para RNAs, sendo o mais difundido o algoritmo de retro-propagação de erro (*error backpropagation*). Treinamento supervisionado é definido como o estímulo de uma entrada x e a saída da rede y é comparada com o valor desejado (valor real). A partir desta diferença os pesos são ajustados até que o resultado desejado seja obtido. Existem diversas RNAs com arquiteturas diferentes da MLP desenvolvidas, podendo-se destacar as Máquinas de Aprendizagem Extremo (*Extreme Learning Machines* - ELM) (HUANG; ZHU; SIEW, 2006), Funções de Base Radial (*Radial Basis Function* - RBF) (POWELL, 1987) e Mapas Auto-Organizados (*Self-organizing Maps* - SOM) (HULLE, 2012).

Figura 2.5 – Representação de uma MLP com duas camadas escondidas.



Fonte: Adaptado de Haykin et al. (2009).

2.5 Algoritmos de Classificação Incrementais (*Online*)

Técnicas de aprendizado não-incrementais têm algumas restrições que podem interferir no desempenho do modelo computacional, sendo estas (FONTENLA-ROMERO et al., 2013):

1. Necessidade de se ter, no momento do treinamento, todo o conjunto de dados para ajuste do modelo preditivo. A cada etapa do processo de aprendizado existe a necessidade do acesso imediato e completo do conjunto de dados.
2. Não pode haver restrições de tempo, isto é, é preciso aguardar o tempo suficiente para o ajuste completo do modelo seja realizado.
3. O processo subjacente aos dados de treinamento não pode sofrer mudanças. Uma vez que o modelo é ajustado, não são necessárias adaptações do mesmo desde que o processo relacionado não sofra mudanças.

Estas restrições supracitadas se aplicam ao problema de identificação de condutores, uma vez que se trata de identificar o modo no qual o condutor opera o veículo. Este processo subjacente pode sofrer mudanças a todo o momento, dependendo de fatores temporais (horário, dia da semana) e climáticos (chuva, nevoeiro), e até mesmo alterações inerentes à estes, como uma mudança de hábitos de direção. Sendo assim, para implementações reais deste sistema, deve-se considerar outro paradigma de aprendizado de máquina para a realização da tarefa de classificação: o Aprendizado Incremental ou *Online*.

Aprendizado de Máquina Incremental é um método dentro de Inteligência Computacional em que os dados tornam-se disponíveis sequencialmente e são usados para a atualização

$$F_1 = 2 \frac{precision \cdot recall}{precision + recall} \quad (2.20)$$

O última medida colhida é a análise de concordância kappa (*Cohen's kappa score*), que varia entre 0 e 1, os quais podem ser interpretados de acordo com a Tabela 2.2. Uma explanação mais completa destas métricas pode ser encontrado em (BANERJEE et al., 1999).

Tabela 2.2 – Interpretação dos valores de Kappa

Kappa	Interpretação
<0	Sem concordância
0-0.19	Concordância baixa
0.20-0.39	Concordância razoável
0.40-0.59	Concordância moderada
0.60-0.79	Concordância substancial
0.80-1.00	Concordância quase perfeita

3 METODOLOGIA

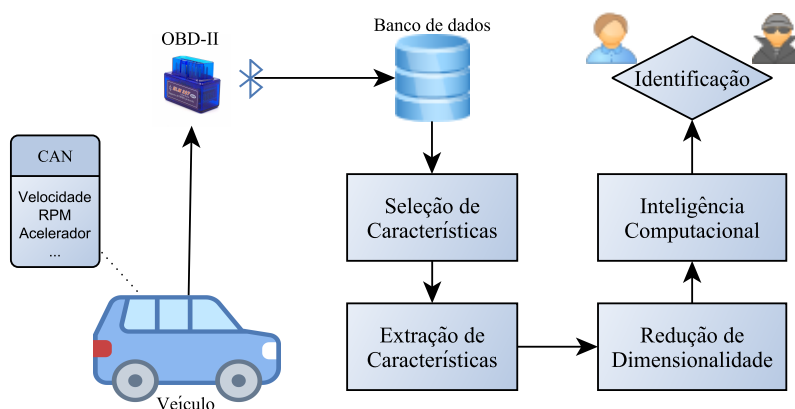
Este capítulo descreve a metodologia e as técnicas que serão empregados na pesquisa e desenvolvimento do sistema proposto, destacando as tecnologias e dispositivos necessários.

3.1 Visão geral do sistema proposto

O Sistema de Identificação de Condutores baseado em Técnicas de Inteligência Computacional terá a função de efetuar a identificação do condutor que dirige o veículo e detectar possíveis situações de roubo e furto do mesmo. Para tal, a identificação será feita por meio do reconhecimento de certos padrões de direção característicos de cada condutor. Isto é possível através da coleta de dados provenientes do barramento de comunicação do veículo, por meio da interface OBD-II e coletados pelo dispositivo de leitura ELM327, que transmite via *bluetooth* para o *smartphone* a ele conectado, que também fornece dados referentes à dinâmica de direção.

Antes de submeter os dados no algoritmo de identificação, será efetuado um processamento e fusão destes dados, de modo a remover ruídos e possíveis *outliers*, além de melhorar a extração de características pelo sistema. Feito isso, dentro do grupo de condutores de cada *dataset*, será determinado um conjunto de condutores autorizados, treinando o modelo computacional com as características destes. Após o treinamento, serão efetuados testes com dados de direção desconhecidos pelo modelo, de forma a avaliar o desempenho do mesmo. Esta validação será utilizada para comprovar a viabilidade do sistema de identificação de condutores. A Figura 3.1 apresenta graficamente o sistema e os passos até a identificação.

Figura 3.1 – Visão geral do Sistema de Identificação de Condutores baseado em Técnicas de Inteligência Computacional.

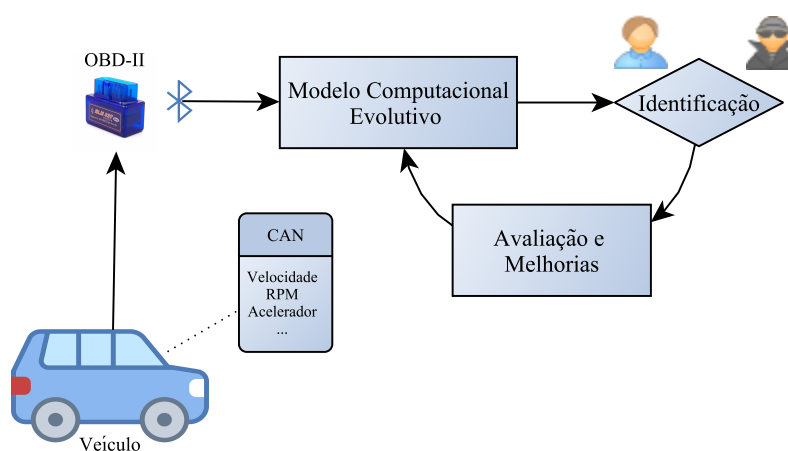


Fonte: Próprio autor.

Uma vez determinada a viabilidade da identificação de condutores por meio dos modelos implementados, será desenvolvido um sistema evolutivo, que poderá ser implementado em aplicações reais. Este modelo terá como objetivo aprimorar seu sistema de classificação através do uso contínuo de dados de direção gerados pelos condutores autorizados, uma vez que o estilo de direção poderá ser alterado por uma série de fatores, como horário, questões climáticas (chuva, neblina, entre outros) e emocionais do condutor.

A Figura 3.2 apresenta o esquema de operação deste sistema. No modelo computacional evolutivo já está incluso as técnicas de seleção e extração de características, bem como as de redução de dimensionalidade (caso necessário). O principal motivo para a implementação de um modelo de aprendizado *online* é justamente a necessidade do modelo estar sempre se aprimorando, baseado no fluxo de dados de direção que constantemente o alimentará. A ideia é que a tarefa de classificação seja sempre a mais precisa possível e que se adapte aos mais diferentes estilos de direção de cada condutor autorizado.

Figura 3.2 – Visão geral do Sistema de Identificação de Condutores baseado em Técnicas de Inteligência Computacional.



Fonte: Próprio autor.

3.2 Datasets Utilizados

No desenvolvimento preliminar do sistema proposto foi utilizado o *dataset* disponibilizado por Kwak, Woo e Kim (2016), juntamente com o *Hacking and Countermeasure Research Lab*¹. Este *dataset* consiste em dados extraídos do barramento CAN do veículo, coletados por

¹ <http://ocslab.hksecurity.net>

por meio da interface *On Board Diagnostics 2* (OBD-II) do veículo e do *scanner* CarbigP. O veículo usado na coleta dos dados (Kia Soul) possui diversos sensores e atuadores que fornecem um montante representativo de dados que podem ser usados para detectar certos padrões inerentes ao comportamento do condutor e até mesmo sua identidade. Estes dados foram amostrados em uma frequência de 1Hz com 51 sensores medidos.

Na coleta de dados, dez indivíduos conduziram o veículo em quatro rotas distintas em Seul, Coreia do Sul. Os autores consideraram três categorias de vias: urbano, rodovia e estacionamento. Cada condutor foi submetido a duas sessões de direção que passa por todos os tipos de vias, totalizando 23 km.

A grande vantagem desse conjunto de dados em comparação à outros utilizados em abordagens similares é que este *dataset* é mais recente, colhido a partir de julho de 2015, e mais completo uma vez que veículos mais novos possuem maior quantidade de sensores embarcados. Outra vantagem relevante do mesmo é a preocupação que os autores tiveram em relação à confiabilidade dos dados, visto que os experimentos foram conduzidos em situações similares, como o fator temporal (entre 8 da manhã e 11 da noite em dias de semana) e duas viagens diferentes para cada condutor, aumentando a extração de traços característicos de direção de cada condutor, e portanto, proporcionando uma classificação de condutores mais eficiente. O conjunto de dados fornecidos tem um total de 94.401 amostras a 1Hz, com tamanho total de 16,7Mb.

Em testes futuros também será considerado o uso do *dataset* UYANIK, cordialmente cedido pelo VPALAB da Universidade de Sabanci (Istambul, Turquia) (ABUT et al., 2007). Este *dataset* contém dados de um grande número de condutores colhidos a partir de dados do barramento CAN, sensores inerciais e GPS e é amplamente utilizado em diversos trabalhos que envolvem identificação de condutores (MARTINEZ; ECHANOBÉ; CAMPO, 2016) (JAFARNEJAD; CASTIGNANI; ENGEL, 2017) (CAMPO et al., 2014). Estes dados servirão para avaliar a aptidão das técnicas de processamento de dados e de inteligência computacional, de modo a se determinar as que melhores se encaixam na tarefa de identificação e autenticação dos condutores. Entre as vantagens deste conjunto de dados é o fato de, além dos dados do barramento CAN, estar presente dados obtidos a partir de sensores inerciais (tal qual encontrado nos *smartphones* atuais), que proporcionam informações mais completas relacionadas à dinâmica de direção.

Outro *dataset* que poderá ser utilizado é o *Driving Behavior Dataset* do LAC (*Laboratory for Advanced Collaboration*)¹ da PUC-Rio, sendo este o único banco de dados de condução brasileiro (Oliveira Vasconcelos, 2017). O veículo utilizado foi a versão brasileira do Citroën C3, equipado com um *smartphone* conectado a um dispositivo OBD-II via *bluetooth*. Foram selecionados 25 motoristas para a coleta dos dados, sendo dezesseis do sexo masculino e nove do sexo feminino, com idade variando de 20 a 60 anos. Os motoristas possuíam de 2 a 42 anos de experiência. Além dos dados do OBD-II, foram aproveitados os sensores embarcados no *smartphone*, sendo estes: acelerômetro, GPS, giroscópio e orientação (bússula). Foi definido um percurso pavimentado em Aracaju-SE totalizando aproximadamente 14.5 Km, cujas ruas e avenidas possuíam de uma a três faixas. Além disso, a rota contém rotatórias, semáforos, faixa de pedestres, cruzamentos, interseções (parada total) e curvas de aproximadamente 45° e 90°. O limite de velocidade em todo o percurso é de 60 Km/h.

3.3 Desenvolvimento Preliminar

Com o objetivo de analisar e comprovar a viabilidade do sistema proposto, foram elaborados uma série de experimentos preliminares, analisando diversas técnicas de seleção e extração de características, bem como diferentes algoritmos de classificação. Dentre os *datasets* supracitados, foi utilizado o fornecido pelo *Hacking and Countermeasure Research Lab*.

3.3.1 Seleção de Características

O conjunto de dados é dividido em dez condutores numerados entre 0 e 9. Porém o objetivo do trabalho é classificar os condutores entre não autorizado e autorizado. Para tal, foram selecionados aleatoriamente três condutores que foram definidos como autorizados (classe 1), como membros de uma mesma família ou empresa que compartilham o mesmo veículo, e restante dos condutores foram definidos como não autorizados (classe 0).

A implementação do sistema de testes passa inicialmente pela seleção de quais características disponíveis no conjunto de dados colhidos no barramento CAN serão utilizadas na tarefa de identificação dos condutores. Neste estágio, foi utilizado a técnica FDA, implementado para Python por Li et al. (2016) especialmente para seleção de características para classificadores. Foram selecionadas oito características de entrada de maior relevância, as quais são descritas na Tabela 3.1.

¹ <http://www.inf.puc-rio.br/rvasconcelos/DataSet/dataset.html>

Tabela 3.1 – Características de entrada selecionadas.

Característica	Descrição
1	Valor do pedal de acelerador
2	Ângulo de esterçamento do volante
3	Velocidade do Veículo
4	Velocidade do Motor
5	Abertura da Válvula reguladora
6	Aceleração Longitudinal
7	Aceleração Lateral
8	Pressão do ar de entrada

3.3.2 Extração de Características

De posse destas variáveis, passa-se agora ao estágio de extração de características de forma a minimizar os efeitos de flutuação dos valores e caracterizar a distribuição de características efetivamente. Além dos valores de cumulantes em atraso zero (variância (segunda ordem), assimetria (terceira ordem) e curtose (quarta ordem)), foram utilizados os valores de mediana e média, totalizando assim cinco características estatísticas por variável.

Estas estatísticas foram obtidas por meio de certo período de amostragem determinadas por meio de um janelamento temporal. De forma a também determinar qual o intervalo de tempo de janelamento é mais adequado para a tarefa de identificação dos condutores, foram testadas janelas de 15, 30, 60, 90 e 120 segundos.

3.3.3 Redução da Dimensionalidade

Pode-se observar que no estágio de extração de características estatística, de cada variável de entrada são derivados cinco elementos, totalizando quarenta variáveis de entrada. Esta quantidade de valores dificulta o desempenho do classificador em termos de eficiência computacional e até comprometer a capacidade de generalização do mesmo. Sendo assim, mais uma vez foi aplicado o Discriminante de Fisher de forma a selecionar destas quarenta variáveis derivadas as dez mais significantes para a tarefa de classificação.

Ainda com a redução obtida no passo anterior, o número de variáveis de entrada pode dificultar o desempenho do classificador. Para contornar esta situação, foram adotados métodos de redução de dimensionalidade de forma a reduzir o espaço de entrada de dez variáveis para somente quatro, sem que ocorram perdas de características significativas. As técnicas testadas são as de Análise de Componentes Principais (PCA), Análise de Componentes Principais Incrementais (IPCA) e Análise de Componentes Independentes (ICA), todos considerando quatro

componentes. Portanto o espaço de dez variáveis oriundos da seleção por meio do Discriminante de Fisher foi reduzido a um espaço de apenas quatro componentes no classificador.

3.3.4 Classificação

Para a tarefa de identificação dos condutores foram considerados três algoritmos de classificação distintos. Os algoritmos foram implementados em Python utilizando a biblioteca *scikit-learn - Machine Learning in Python* (PEDREGOSA et al., 2011). Os classificadores foram configurados de acordo com os seguintes parâmetros:

- K – *Nearest Neighbor* (KNN): número de vizinhos: 3; pesos uniformes.
- *Random Forest* (RF): 50 estimadores; profundidade máxima: 10; amostras mínimas por folha: 10.
- Redes Neurais Artificiais (MLP): Número de camadas escondidas = 2, número de neurônios por camada = 10, máximo de épocas de treinamento = 500.

3.3.5 Configuração dos experimentos

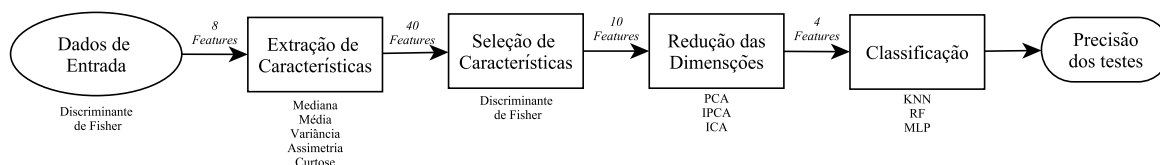
Os experimentos foram conduzidos de forma a testar todas as combinações de técnicas possíveis de forma a se conseguir o maior número de acertos na identificação dos condutores. Estas combinações envolvem as cinco janelas temporais (15, 30, 60, 90 e 120 segundos), três técnicas de redução de dimensionalidade (PCA, IPCA e ICA) e três classificadores (KNN, RF e MLP). A combinação destes elementos totalizam quarenta e cinco cenários diferentes. Cada classificador é treinado e testado dez vezes e cada uma destas é efetuada a separação entre dados de treino (80%) e teste (20%). Em cada um dos testes é feita uma permutação aleatória dos dados, de forma a não comprometer a capacidade de generalização do classificador.

O objetivo final dos testes é maximizar a taxa de acertos entre os condutores autorizados (classe 1) e não autorizados (classe 0). O fluxo geral dos experimentos é descrito na Figura 3.3.

3.4 Modelo Computacional Evolutivo

Uma vez desenvolvido e testado o modelo computacional preliminar com o objetivo de comprovar a viabilidade do sistema de identificação de condutores proposto, será tomado como passo subsequente o desenvolvimento de um modelo computacional que trabalhe de modo a

Figura 3.3 – Fluxo de desenvolvimento e testes do sistema preliminar.



Fonte: Próprio autor.

evoluir a medida em que o mesmo seja submetido a um fluxo de dados de direção. Para tanto, será avaliado técnicas de aprendizado de máquina *online*, ou seja, que sejam capazes de se adaptar a diferentes tipos de situações e que não dependa de um novo treinamento com massa de dados.

Muitos trabalhos que buscam realizar a modelagem de comportamento de condutores com diversas finalidades, apesar de desempenhar detecção de anomalias de forma *online* (JIAN-QIANG; YI-YING, 2014) Zhang et al. (2016), não exploram algoritmos evolutivos propriamente ditos, uma vez que somente a configuração dos classificadores parte do pressuposto que o processo subjacente é imutável. Sendo assim, serão avaliados algoritmos de aprendizado *online* que tenham desempenho satisfatório à tarefa de classificação e que também sejam capazes de evoluir seu modelo a medida que novos dados são gerados. Um grupo de técnicas serão selecionadas e testadas tal que a que com o melhor desempenho e com aprendizado mais eficiente seja utilizada em implementações futuras.

3.4.1 Configuração de Experimentos Futuros

Os experimentos para avaliar o modelo de aprendizado *online* serão elaborados para que os dados disponíveis nos *datasets* descritos sejam submetidos no modelo computacional de forma sequencial e ordenada, simulando assim um operação direção. Assim como explorado para técnicas de aprendizado *batch*, um janelamento temporal será avaliado em conjunto com técnicas de extração e seleção de características de forma a maximizar o desempenho do classificador. Outro ponto a ser explorado é a fusão de dados de diferentes sensores complementares, como os inerciais, que estão disponíveis nos *datasets* fornecidos por Abut et al. (2007) e Oliveira Vasconcelos (2017).

4 RESULTADOS PRELIMINARES E ESPERADOS

4.1 Resultados Preliminares

Uma etapa determinante no desenvolvimento do sistema de identificação de condutores é a comprovação de viabilidade do mesmo. Para tal foram executados experimentos com classificadores *batch* em conjunto com métodos de processamento de dados. O objetivo dos testes é de se obter um valor alto de precisão (acertos entre condutores autorizados e não autorizados). A configuração destes experimentos está explanada no Capítulo 3 do presente projeto de pesquisa. Os gráficos da Figura 4.1 mostram o desempenho em termos da acurácia dos testes com a dispersão dos resultados dos classificadores, divididos em cada umas das técnicas de redução de dimensionalidade.

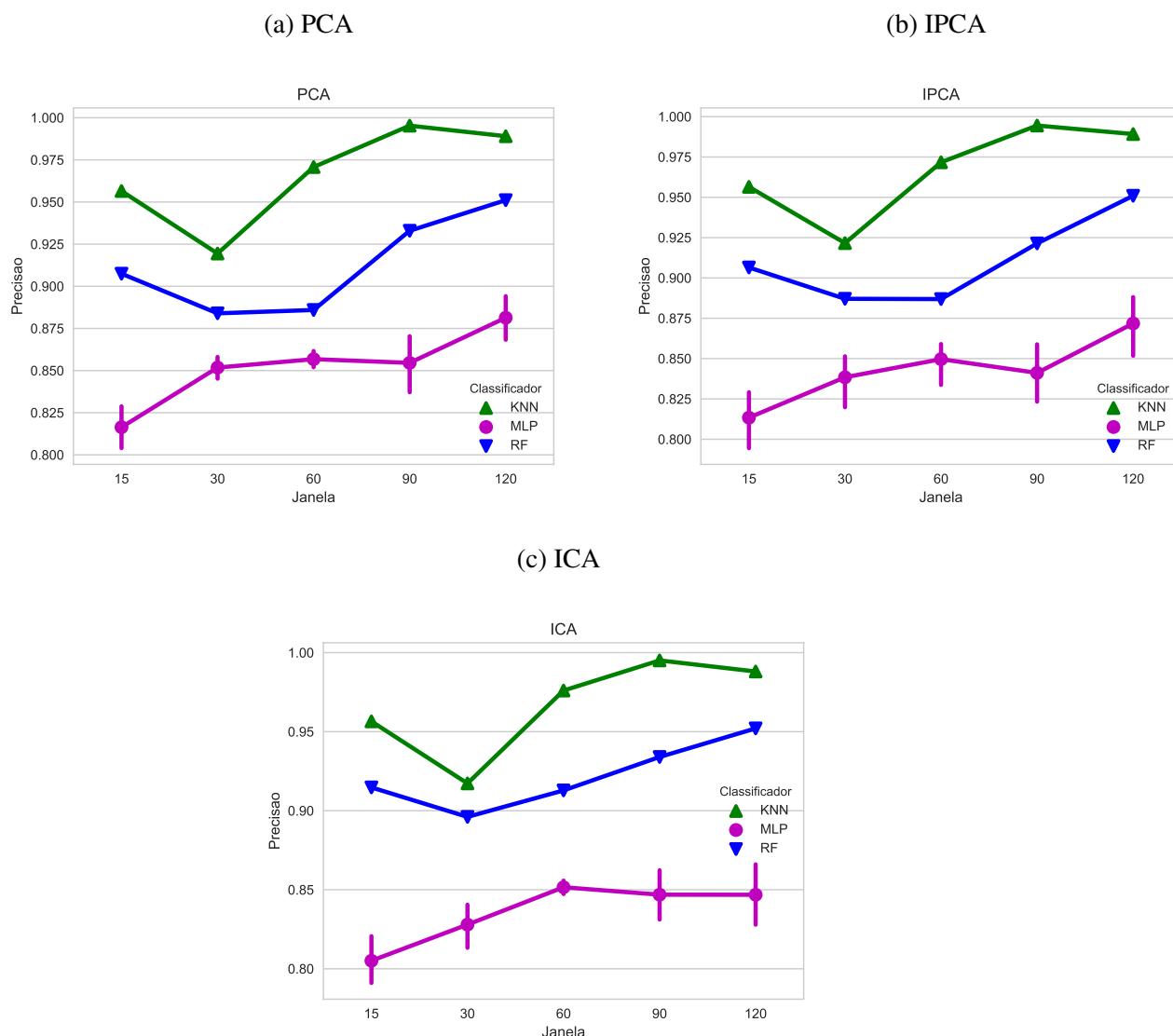
Em todos os cenários, o classificador KNN obteve desempenho superior em comparação aos outros classificadores, enquanto o RNA teve um desempenho consideravelmente inferior, além de apresentar valores de desvio padrão superiores, se mostrando uma técnica pouco indicada para a tarefa de identificação de condutores.

Considerando a janela temporal ideal para a identificação de condutores, conclui-se que a janela de 90 segundos maximiza o desempenho do classificador KNN, porém, nos casos das técnicas RNA e RF, quanto maior a janela temporal, melhor a taxa de acertos dos mesmos. A janela temporal de 30 segundos apresentou desempenho inferior no KNN e RF, enquanto no caso do RNA, os valores se estabilizaram a partir da janela de 30 segundo, com acréscimo razoável na janela de 120 segundos.

Pode-se observar que a janela de 90 segundos tem o melhor desempenho no que diz respeito aos certos quanto à autenticidade do condutor. É válido ressaltar que, em aplicações reais futuras, deve-se determinar um valor de janela ótimo que alie a rapidez e precisão na identificação do condutor, visto que uma janela de 90 segundo pode ser muito quanto se trata de uma ação criminosa, porém, um falso negativo é uma situação que deve ser evitada para a confiabilidade do sistema.

Com relação à eficiência dos métodos de redução de dimensionalidade, nenhuma das técnicas se mostrou superior em questão de desempenho em comparação às outras, no qual o objetivo de diminuir o espaço amostral de entrada do classificador foi alcançado sem que se houvesse comprometimento na classificação.

Figura 4.1 – Precisão média de cada método de redução de dimensionalidade.

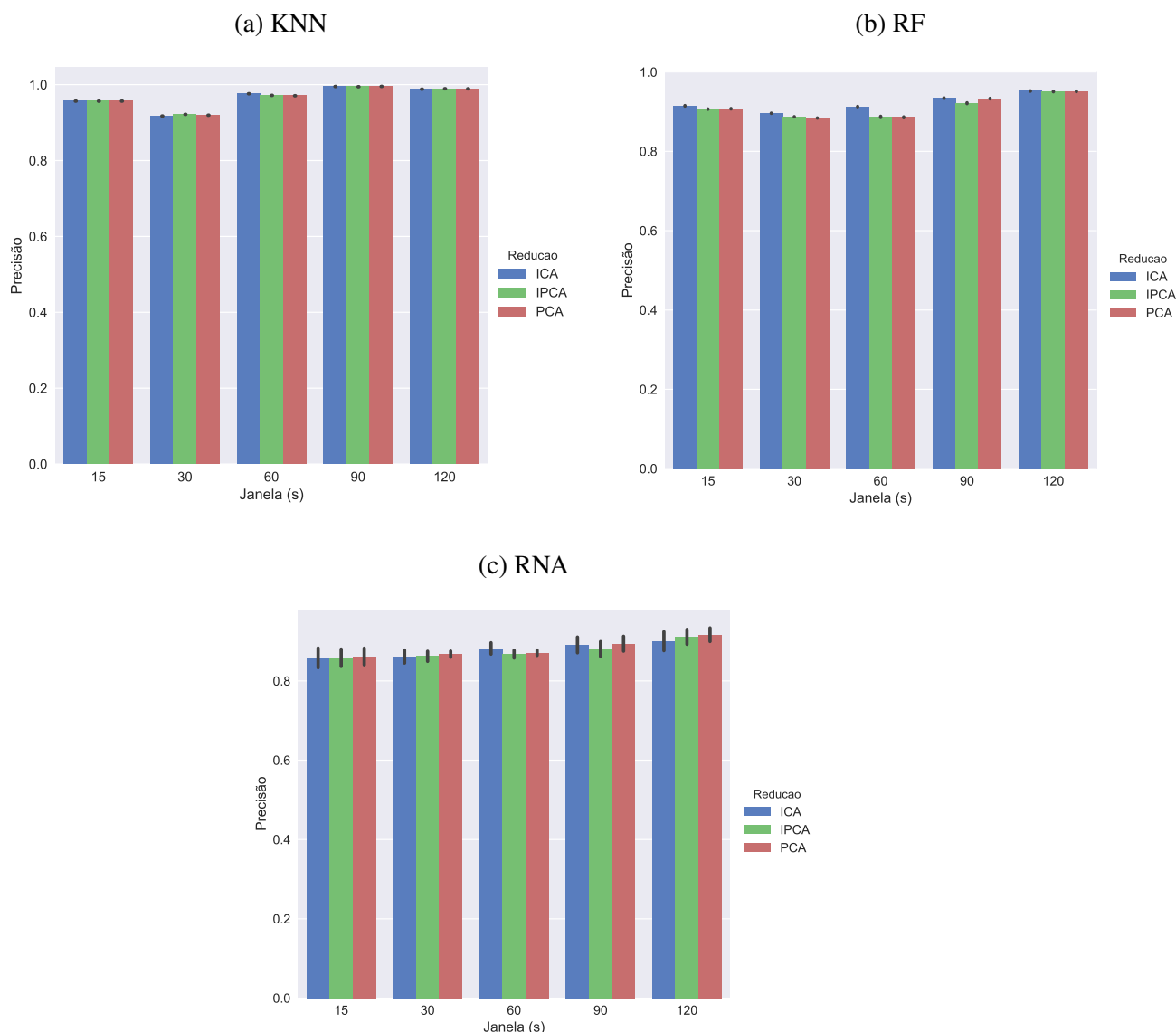


Fonte: Próprio autor.

No caso do melhor classificador, o KNN, as três técnicas obtiveram uma quantidade de acertos semelhantes nas mesmas condições de janela temporal. Contudo, nos classificadores RF e RNA não houve uma técnica que teve seu desempenho superior em todas as circunstâncias. Os gráficos apresentados na Figura 4.2 ilustram o desempenho comparado de cada uma das técnicas de redução de dimensionalidade em cada um dos classificadores abordados.

A Tabelas 4.1, 4.2 e 4.3 apresentam uma análise de desempenho mais aprofundada dos classificadores, considerando, além da precisão, as métricas de análise de concordância *Cohen's kappa Score* e *F1-Score*. Para o classificador kNN o valor do desvio padrão foi insignificante e, portanto, foi desconsiderado. No caso do classificador kNN, pode-se observar que em todos os casos, a janela de 90 segundos resulta em um resultado melhor chegando à 99.5% de acertos

Figura 4.2 – Precisão média de cada classificador por método de redução de dimensionalidade.



Fonte: Próprio autor.

com redução PCA e ICA e 99.4% na redução IPCA. Por sua vez, o classificador *Random Forest* A janela de 120 segundos foi a que obteve o melhor desempenho, chegando precisão média chegando à 95.2%, com concordância de 88.4%, mostrando que quanto mais tempo usado no janelamento, melhor o seu desempenho, o que pode ser um ponto negativo, devido ao tempo excessivo para o início da identificação dos condutores. Já a RNA teve um desempenho inferior aos demais chegando na precisão máxima de 88.1% com apenas de 71.3% de concordância (considerando o desequilíbrio das classes).

Diante dos resultados apresentados nesta seção, pode-se observar que o sistema de identificação de condutores com classificadores (*batch*) se mostrou eficaz na maioria dos experi-

Tabela 4.1 – Desempenho do Classificador KNN e Técnicas de Redução de Dimensionalidade em diferentes métricas

Janela (s)	kNN								
	PCA			IPCA			ICA		
	Precisão	F1-Score	Cohen Kappa	Precisão	F1-Score	Cohen Kappa	Precisão	F1-Score	Cohen Kappa
15	0.956	0.927	0.896	0.956	0.927	0.896	0.957	0.927	0.896
30	0.919	0.838	0.785	0.922	0.842	0.791	0.917	0.834	0.778
60	0.971	0.951	0.930	0.972	0.952	0.932	0.976	0.960	0.943
90	0.995	0.992	0.988	0.994	0.990	0.986	0.995	0.991	0.988
120	0.989	0.981	0.974	0.989	0.981	0.974	0.988	0.980	0.971

Tabela 4.2 – Desempenho do Classificador *Random Forest* e Técnicas de Redução de Dimensionalidade em diferentes métricas

Janela (s)	Random Forest								
	PCA			IPCA			ICA		
	Precisão	F1-Score	Cohen Kappa	Precisão	F1-Score	Cohen Kappa	Precisão	F1-Score	Cohen Kappa
15	0.907±0.001	0.841±0.001	0.775±0.002	0.906±0	0.839±0.001	0.773±0.001	0.913±0.001	0.850±0.002	0.791±0.003
30	0.884±0	0.730±0.001	0.659±0.001	0.887±0	0.737±0.002	0.668±0.002	0.896±0	0.765±0.001	0.701±0.002
60	0.889±0.002	0.789±0.005	0.712±0.006	0.887±0.003	0.789±0.005	0.713±0.007	0.913±0.001	0.846±0.002	0.786±0.002
90	0.933±0.001	0.870±0.002	0.825±0.003	0.921±0.002	0.844±0.004	0.792±0.005	0.934±0.002	0.872±0.004	0.828±0.004
120	0.951±0.001	0.915±0.002	0.881±0.003	0.951±0.001	0.915±0.002	0.880±0.003	0.952±0.001	0.918±0.002	0.884±0.002

Tabela 4.3 – Desempenho do Classificador RNA e Técnicas de Redução de Dimensionalidade em diferentes métricas

Janela (s)	RNA								
	PCA			IPCA			ICA		
	Precisão	F1-Score	Cohen Kappa	Precisão	F1-Score	Cohen Kappa	Precisão	F1-Score	Cohen Kappa
15	0.816±0.021	0.678±0.044	0.550±0.057	0.8134±0.028	0.661±0.078	0.535±0.090	0.805±0.026	0.642±0.065	0.510±0.079
30	0.852±0.011	0.630±0.045	0.546±0.046	0.838±0.026	0.592±0.088	0.501±0.096	0.828±0.23	0.538±0.091	0.450±0.092
60	0.857±0.008	0.734±0.031	0.638±0.032	0.850±0.024	0.718±0.103	0.620±0.098	0.851±0.007	0.735±0.016	0.633±0.020
90	0.854±0.028	0.710±0.060	0.615±0.078	0.841±0.029	0.680±0.060	0.577±0.078	0.847±0.026	0.696±0.052	0.596±0.070
120	0.881±0.022	0.797±0.042	0.713±0.057	0.872±0.032	0.775±0.068	0.686±0.088	0.847±0.032	0.719±0.078	0.615±0.096

mentos. A escolha do classificador também é um fator determinante para o sucesso da tarefa de identificação, onde, dentre todas as técnicas implementadas, o algoritmo de classificação foi o que mais teve impacto no desempenho. Contudo, diante dos resultados satisfatórios obtidos, comprova-se a viabilidade do sistema de identificação de condutores.

4.2 Resultados esperados

Espera-se com os resultados desta pesquisa, desenvolver um sistema, que, através de técnicas de inteligência computacional, possa identificar, autenticar e discernir condutores através de sinais presentes no barramento CAN e sensores inerciais, tais como encontrados em *smartphones*, de forma que este sistema evolua a medida que seja usado. Para tanto, como produto final desta pesquisa, espera-se o desenvolvimento de um modelo computacional evolutivo que desempenhe as tarefas mencionadas, com uma alta taxa de acertos na autenticação dos condutores. Também espera-se a publicação de um ou mais artigos ou resumos expandidos em periódicos ou congressos da área de Engenharias IV.

5 EQUIPE

A equipe de pesquisadores que irá trabalhar na pesquisa e desenvolvimento no presente projeto está descrito a seguir:

- Dr. Wilian Soares Lacerda (Orientador): é Técnico em Eletrônica pelo Centro Federal de Educação Tecnológica de Minas Gerais - CEFET/MG (1986), possui graduação em Engenharia Elétrica pela Universidade Federal de Minas Gerais (1991), mestrado em Engenharia Elétrica (área Automática) pela Universidade Federal de Minas Gerais (1994) e doutorado em Engenharia Elétrica (linha Engenharia da Computação) pela Universidade Federal de Minas Gerais (2006). Atualmente é professor associado da Universidade Federal de Lavras atuando no Departamento de Ciência da Computação onde leciona as disciplinas de graduação e pós-graduação: Eletrônica Básica, Sistemas Digitais, Sistemas Embarcados e Microcontroladores, Redes Neurais Artificiais. Desenvolve pesquisa na área de Inteligência Computacional, atuando principalmente nos seguintes temas: Redes Neurais Artificiais, Sistemas Fuzzy e Computação Evolutiva. Desenvolve protótipos de sistemas embarcados para aplicações específicas utilizando: microcontroladores, *hardware* reconfigurável (FPGA), sensores e atuadores diversos.
- Andrey Gustavo de Souza (Mestrando): possui graduação em Engenharia Elétrica pelo Instituto Federal de Minas Gerais (IFMG) - Campus Formiga. É aluno de mestrado em Engenharia de Sistemas e Automação, na linha de pesquisa em Sistemas Inteligentes, pela Universidade Federal de Lavras (UFLA). É ex-bolsista do programa Ciência sem Fronteiras nos Estados Unidos, tendo feito imersão de inglês na State University of New York (SUNY), Nassau Community College e programa acadêmico na faculdade de engenharia Elétrica da SUNY Maritime College, participando de projetos de pesquisa com o corpo docente da instituição. Atuou como bolsista em projetos de Iniciação Científica dentro do IFMG - Campus Formiga. Têm interesse nas áreas de Sistemas Embarcados, Sistemas Automotivos, Inteligência Computacional e Processamento de Dados.

Poderá-se avaliar a necessidade da inclusão de outros membros à equipe de pesquisa, dentre coorientadores e alunos de iniciação científica, de acordo com as demandas identificadas no andamento do projeto.

6 ORÇAMENTO

Para o desenvolvimento desta pesquisa, serão usados *softwares* de uso gratuito (Python) ou que estão disponíveis para uso através do Departamento de Engenharia (DEG) da Universidade Federal de Lavras (UFLA) (MATLAB). Os equipamentos que serão necessários para a implementação, bem como seus valores, já foram adquiridos ou estão disponíveis pelos Laboratórios da UFLA, conforme descrito na Tabela 6.1.

Tabela 6.1 – Equipamentos necessários e orçamento para desenvolvimento do projeto.

Equipamento	Fonte/Proprietário	Valor (R\$)
Microcomputador	Pesquisador/LABSINE (UFLA)	3.000,00
Software Python e Bibliotecas	Anaconda, Inc.	0,00
MATLAB	UFLA	2,000.00
Total		5.000,00

Fonte: Próprio autor.

7 CRONOGRAMA

Com início no primeiro período de 2017 (maio), o programa de Mestrado se iniciou com a escolha do tema da dissertação, sendo definido o título do presente projeto de pesquisa. Juntamente com o orientador, foram definidas as disciplinas que seriam cursadas. No primeiro período de 2017 foram cursadas as disciplinas: PCH501 Inglês Instrumental, PSI505 Tópicos Especiais em Engenharia de Sistemas e Automação (Redes Neurais Artificiais), PSI515 Pesquisa Bibliográfica e Comunicação Científica, PSI528 Processamento de Sinais e PSI531 Sistemas Fuzzy. No segundo período de 2017 foram cursadas as seguintes disciplinas: PSI504 Seminário I, PSI511 Estágio Docência I MS, PSI514 Projeto Orientado, PSI516 Computação Evolucionária, PSI519 Análise de Componentes Independentes e PSI535 Eletrônica de Potência Aplicada a Sistemas Elétricos. Estas disciplinas integralizam a carga horária mínima obrigatória de 24 créditos.

Dando continuidade ao trabalho que resultará na dissertação de mestrado, é determinado o cronograma apresentado na Tabela 7.1, com as seguintes atividades:

1. Início do mestrado.
2. Cumprimento da carga horária mínima exigida em disciplinas.
3. Revisão da literatura, construção de acervo bibliográfico e análise do estado da arte do sistema proposto.
4. Escrita do Projeto de Qualificação.
5. Avaliação e implementação de testes preliminares de métodos de processamento de dados e algoritmo de identificação de condutores.
6. Avaliação da viabilidade do sistema de identificação de condutores por meio dos resultados preliminares.
7. Exame de Qualificação.
8. Avaliação de técnicas de aprendizado *online* a serem implementadas.
9. Implementação e testes com as técnicas escolhidas e definição da técnica a ser utilizada no sistema.

10. Teste e validação do sistema proposto, através de experimentos e avaliação dos resultados.
11. Escrita da dissertação.
12. Defesa da dissertação.

Tabela 7.1 – Cronograma das atividades que serão realizadas.

Atividade	2017								2018												2019	
	Maio	Jun.	Jul.	Ago.	Set.	Out.	Nov.	Dez.	Jan.	Fev.	Mar.	Abr.	Maio	Jun.	Jul.	Ago.	Set.	Out.	Nov.	Dez.	Jan.	Fev.
1	X																					
2	X	X	X	X	X	X	X	X	X	X												
3					X	X	X	X	X	X	X											
4							X	X	X	X	X	X										
5						X	X	X	X	X	X											
6												X										
7													X									
8														X	X	X						
9																X	X					
10																		X	X	X		
11															X	X	X	X	X	X	X	
12																						X

Fonte: Próprio autor.

REFERÊNCIAS

- ABUT, H. et al. Data collection with uyanik: Too much pain; but gains are coming. In: **Proc. of the Biennial on DSP for In-Vehicle and Mobile Systems**. [S.l.: s.n.], 2007.
- ARTAC, M.; JOGAN, M.; LEONARDIS, A. Incremental pca for on-line visual learning and recognition. In: **Object recognition supported by user interaction for service robots**. [S.l.: s.n.], 2002. v. 3, p. 781–784 vol.3. ISSN 1051-4651.
- BALSUBRAMANI, A.; DASGUPTA, S.; FREUND, Y. The fast convergence of incremental pca. In: BURGESS, C. J. C. et al. (Ed.). **Advances in Neural Information Processing Systems 26**. [S.l.]: Curran Associates, Inc., 2013. p. 3174–3182.
- BANERJEE, M. et al. Beyond kappa: A review of interrater agreement measures. **Canadian journal of statistics**, Wiley Online Library, v. 27, n. 1, p. 3–23, 1999.
- BARBOSA, H. C. **Conceito de dispositivos registradores de dados para veículos inteligentes**. Dissertação (Mestrado em Engenharia de Sistemas e Automação) — Universidade Federal de Lavras, 2017.
- BLASZCZYK, P.; TUREK, W.; CETNAROWICZ, K. Extensible platform for studying the behavior of drivers in urban traffic. In: **2014 17th IEEE International Conference on Intelligent Transportation Systems, ITSC 2014**. [S.l.]: IEEE, 2014. p. 1359–1362. ISBN 9781479960781.
- BREIMAN, L. Random forests. **UC Berkeley TR567**, 1999.
- BREIMAN, L. Random forests. **Machine learning**, Springer, v. 45, n. 1, p. 5–32, 2001.
- Business Analytics. **Introduction to Online Machine Learning : Simplified**. 2015. 1–16 p. Disponível em: <<https://www.analyticsvidhya.com/blog/2015/01/introduction-online-machine-learning-simplified-2/>>.
- CAMPO, I. D. et al. A real-time driver identification system based on artificial neural networks and cepstral analysis. In: **Proceedings of the International Joint Conference on Neural Networks**. [S.l.]: IEEE, 2014. p. 1848–1855. ISSN 2161-4393.
- CAO, L. et al. A comparison of pca, kpca and ica for dimensionality reduction in support vector machine. **Neurocomputing**, Elsevier, v. 55, n. 1, p. 321–336, 2003.
- CARMONA, J. et al. Data Fusion for Driver Behaviour Analysis. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 15, n. 10, p. 25968–25991, oct 2015. ISSN 1424-8220.
- CASTIGNANI, G. et al. Driver behavior profiling using smartphones: A low-cost platform for driver monitoring. **IEEE Intelligent Transportation Systems Magazine**, v. 7, n. 1, p. 91–102, 2015. ISSN 19391390.
- CHIANG, L. H.; KOTANCHEK, M. E.; KORDON, A. K. Fault diagnosis based on fisher discriminant analysis and support vector machines. **Computers & chemical engineering**, Elsevier, v. 28, n. 8, p. 1389–1401, 2004.
- De Winter, J. C. et al. **Effects of adaptive cruise control and highly automated driving on workload and situation awareness: A review of the empirical evidence**. [S.l.]: Pergamon, 2014. 196–217 p.

EBERT, C.; JONES, C. Embedded software: Facts, figures, and future. **Computer**, v. 42, n. 4, p. 42–52, apr 2009. ISSN 00189162.

ECHANOBE, J.; CAMPO, I. del; MARTINEZ, M. V. Design and optimization of a Neural Network-based driver recognition system by means of a multiobjective genetic algorithm. In: **2016 International Joint Conference on Neural Networks (IJCNN)**. [S.l.]: IEEE, 2016. p. 3745–3750. ISBN 978-1-5090-0620-5.

FONTENLA-ROMERO, O. et al. Online machine learning. **Efficiency and Scalability Methods for Computational Intellect**, Hershey, PA: IGI Publishing, v. 27, 2013.

Fórum Brasileiro de Segurança Pública. **11º Anuário Brasileiro de Segurança Pública**. 2017. Disponível em: <<http://www.forumseguranca.org.br/atividades/anuario/>>. Acesso em: 04/11/2017.

GUEDES, J. D.; FERREIRA, D. D.; BARBOSA, B. H. A non-intrusive approach to classify electrical appliances based on higher-order statistics and genetic algorithm: a smart grid perspective. **Electric Power Systems Research**, Elsevier, v. 140, p. 65–69, 2016.

HAYKIN, S. S. et al. **Neural networks and learning machines**. [S.l.]: Pearson Upper Saddle River, NJ, USA:, 2009. v. 3.

HOEGGER, R. et al. Highly Automated Vehicles for Intelligent Transport. **Proceedings of the ITS World Congress**, vol. 1, n. July 2015, p. pp. 1–10, 2009.

HUANG, G.-B.; ZHU, Q.-Y.; SIEW, C.-K. Extreme learning machine: theory and applications. **Neurocomputing**, Elsevier, v. 70, n. 1-3, p. 489–501, 2006.

HUBAUX, J. P.; ČAPKUN, S.; LUO, J. The security and privacy of smart vehicles. **IEEE Security and Privacy**, v. 2, n. 3, p. 49–55, may 2004. ISSN 15407993.

HULLE, M. M. V. Self-organizing maps. In: _____. **Handbook of Natural Computing**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. p. 585–622. ISBN 978-3-540-92910-9.

HYVARINEN, A. Fast and robust fixed-point algorithms for independent component analysis. **IEEE transactions on Neural Networks**, IEEE, v. 10, n. 3, p. 626–634, 1999.

INAGAKI, T. Special issue on human-automation coagency. **Cognition, Technology and Work**, Springer-Verlag, v. 14, n. 1, p. 1–2, mar 2012. ISSN 14355558.

JAFARNEJAD, S.; CASTIGNANI, G.; ENGEL, T. Towards a Real-Time Driver Identification Mechanism Based on Driving Sensing Data. In: **20th International Conference on Intelligent Transportation Systems (ITSC)**. [S.l.: s.n.], 2017. p. 7.

JIAN-QIANG, G.; YI-YING, W. Research on online identification algorithm of dangerous driving behavior. In: IEEE. **Intelligent Systems Design and Engineering Applications (ISDEA), 2014 Fifth International Conference on**. [S.l.], 2014. p. 821–824.

KOLDOVSKY, Z.; TICHAVSKY, P.; OJA, E. Efficient variant of algorithm fastica for independent component analysis attaining the cramér-rao lower bound. **IEEE Transactions on neural networks**, IEEE, v. 17, n. 5, p. 1265–1277, 2006.

- KUMTEPE, O.; AKAR, G. B.; YUNCU, E. Driver aggressiveness detection via multisensory data fusion. **EURASIP Journal on Image and Video Processing**, v. 2016, n. 1, p. 5, 2016. ISSN 1687-5281.
- KWAK, B. I.; WOO, J.; KIM, H. K. Know your master: Driver profiling-based anti-theft method. In: **PST 2016**. [S.l.: s.n.], 2016.
- LEE, T.-W. Independent component analysis. In: **Independent Component Analysis**. [S.l.]: Springer, 1998. p. 27–66.
- LI, J. et al. Feature selection: A data perspective. **arXiv preprint arXiv:1601.07996**, 2016.
- MARTINEZ, M.; ECHANOBÉ, J.; CAMPO, I. del. Driver identification and impostor detection based on driving behavior signals. In: **2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)**. [S.l.]: IEEE, 2016. p. 372–378. ISBN 978-1-5090-1889-5.
- MEIRING, G. A. M.; MYBURGH, H. C. A review of intelligent driving style analysis systems and related artificial intelligence algorithms. **Sensors (Switzerland)**, Multidisciplinary Digital Publishing Institute, v. 15, n. 12, p. 30653–30682, dec 2015. ISSN 14248220.
- MENDEL, J. M. Tutorial on higher-order statistics (spectra) in signal processing and system theory: Theoretical results and some applications. **Proceedings of the IEEE**, Ieee, v. 79, n. 3, p. 278–305, 1991.
- MOREIRA, M. G.; FERREIRA, D. D.; DUQUE, C. A. Hos based power quality disturbance detection. 2014.
- NAVES, R.; BARBOSA, B. H.; FERREIRA, D. D. Classification of lung sounds using higher-order statistics: A divide-and-conquer approach. **Computer methods and programs in biomedicine**, Elsevier, v. 129, p. 12–20, 2016.
- Oliveira Vasconcelos, I. **DETECÇÃO MÓVEL E ONLINE DE ANOMALIA EM MÚLTIPLOS FLUXOS DE DADOS: UMA ABORDAGEM BASEADA EM PROCESSAMENTO DE EVENTOS COMPLEXOS PARA DETECÇÃO DE COMPORTAMENTO DE CONDUÇÃO**. Tese (Doutorado) — PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO, Rio de Janeiro, Brazil, mar 2017.
- PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011.
- PETERSON, L. E. K-nearest neighbor. **Scholarpedia**, v. 4, n. 2, p. 1883, 2009.
- POWELL, M. Algorithms for approximation: Radial basis functions for multivariable interpolation (a review). Clarendon Press, 1987.
- RENESAS. **Advanced Driver Assistance System (ADAS)**. 2017. Disponível em: <www.renesas.com/pt-br/solutions/automotive/adas.html>. Acesso em: 04/11/2017.
- ROSENBLATT, F. Principles of neurodynamics. Spartan Book, 1962.
- SAE INTERNATIONAL. **AUTOMATED DRIVING LEVELS OF DRIVING AUTOMATION ARE DEFINED IN NEW SAE INTERNATIONAL STANDARD J3016**. 2014.

SAFAVIAN, S. R.; LANDGREBE, D. A survey of decision tree classifier methodology. **IEEE transactions on systems, man, and cybernetics**, IEEE, v. 21, n. 3, p. 660–674, 1991.

SAITO, Y.; ITOH, M.; INAGAKI, T. Driver Assistance System with a Dual Control Scheme: Effectiveness of Identifying Driver Drowsiness and Preventing Lane Departure Accidents. **IEEE Transactions on Human-Machine Systems**, v. 46, n. 5, p. 660–671, oct 2016. ISSN 21682291.

SUGIYAMA, M. Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis. **Journal of machine learning research**, v. 8, n. May, p. 1027–1061, 2007.

TUOHY, S. et al. Intra-Vehicle Networks: A Review. **IEEE Transactions on Intelligent Transportation Systems**, v. 16, n. 2, p. 534–545, apr 2015. ISSN 1524-9050.

WANG, J.; CHANG, C.-I. Independent component analysis-based dimensionality reduction with applications in hyperspectral image analysis. **IEEE transactions on geoscience and remote sensing**, IEEE, v. 44, n. 6, p. 1586–1600, 2006.

WENG, J.; ZHANG, Y.; HWANG, W.-S. Candid covariance-free incremental principal component analysis. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 25, n. 8, p. 1034–1040, Aug 2003. ISSN 0162-8828.

WOLD, S.; ESBENSEN, K.; GELADI, P. Principal component analysis. **Chemometrics and intelligent laboratory systems**, Elsevier, v. 2, n. 1-3, p. 37–52, 1987.

ZHANG, H. et al. Image classification using rapideye data: Integration of spectral and textual features in a random forest classifier. **IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing**, v. 10, n. 12, p. 5334–5349, Dec 2017. ISSN 1939-1404.

ZHANG, J. et al. Safety awareness online detection system of driving behavior based on software and hardware co-design. In: IEEE. **Software Quality, Reliability and Security Companion (QRS-C), 2016 IEEE International Conference on**. [S.l.], 2016. p. 395–399.