UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CÂMPUS CORNÉLIO PROCÓPIO
DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

MARDLLA DE SOUSA SILVA

ESTRATÉGIAS DE APRENDIZADO VISANDO MELHORIAS NOS PROCESSOS DE CLASSIFICAÇÃO E DE CONTROLE DE QUALIDADE NA INDÚSTRIA DO RAMO ALIMENTÍCIO

DISSERTAÇÃO – MESTRADO

CORNÉLIO PROCÓPIO

2020

MARDLLA DE SOUSA SILVA

ESTRATÉGIAS DE APRENDIZADO VISANDO MELHORIAS NOS PROCESSOS DE CLASSIFICAÇÃO E DE CONTROLE DE QUALIDADE NA INDÚSTRIA DO RAMO ALIMENTÍCIO

Dissertação apresentada ao Programa de Pós-Graduação em Informática da Universidade Tecnológica Federal do Paraná – UTFPR como requisito parcial para a obtenção do título de "Mestre Profissional em Informática".

Orientadora: Profa. Dra. Priscila Tiemi Maeda Saito

CORNÉLIO PROCÓPIO

2020

Dados Internacionais de Catalogação na Publicação

S586 Silva, Mardlla de Sousa

Estratégias de aprendizado visando melhorias nos processos de classificação e de controle de qualidade na indústria do ramo alimentício / Mardlla de Sousa Silva. - 2020.
60 f. : il. color. ; 31 cm.

Orientador: Priscila Tiemi Maeda Saito.
Dissertação (Mestrado) – Universidade Tecnológica Federal do Paraná. Programa de Pós-Graduação em Informática, Cornélio Procópio, 2020.
Bibliografia: p. 44-48.

1. Aprendizado do computador. 2. Processamento de imagens. 3. Classificação. 4. Controle de qualidade. 5. Informática – Dissertações. I. Saito, Priscila Tiemi Maeda, orient. II. Universidade Tecnológica Federal do Paraná. Programa de Pós-Graduação em Informática. III. Título.

CDD (22. ed.) 004

Biblioteca da UTFPR - Câmpus Cornélio Procópio
Bibliotecário/Documentalista responsável:
Romeu Righetti de Araujo – CRB-9/1676

"É muito melhor lançar-se em busca de conquistas grandiosas, mesmo expondo-se ao fracasso, do que alinhar-se com os pobres de espírito, que nem gozam muito nem sofrem muito, porque vivem numa penumbra cinzenta, onde não conhecem nem vitória, nem derrota."

Theodore Roosevelt

AGRADECIMENTOS

Agradeço a minha orientadora Prof. Dr. Priscila Tiemi Maeda Saito, pela sabedoria com que me guiou nesta trajetória.

Aos meus colegas de sala.

Ao aluno João Marcelo Tozato que me ajudou em alguns pontos deste trabalho.

Ao meu Gestor profissional Geraldo Pereira Júnior que me incentivou a esse novo desafio.

A Secretaria do Curso, pela cooperação.

Gostaria de deixar registrado também, o meu reconhecimento à minha família, pois acredito que sem o apoio deles seria muito difícil vencer esse desafio.

Enfim, a todos os que por algum motivo contribuíram para a realização desta pesquisa.

RESUMO

SILVA, Mardlla S. ESTRATÉGIAS DE APRENDIZADO VISANDO MELHORIAS NOS PROCESSOS DE CLASSIFICAÇÃO E DE CONTROLE DE QUALIDADE NA INDÚSTRIA DO RAMO ALIMENTÍCIO. 62 f. Dissertação – Mestrado – Programa de Pós-Graduação em Informática, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2020.

Considerando a grande concorrência entre as indústrias, um dos principais fatores que tornam as empresas líderes de mercado é a qualidade de seus produtos. No entanto, as técnicas aplicadas ao controle de qualidade são muitas vezes falhas ou ineficientes, devido à grande dependência do fator humano, o que torna os procedimentos aplicados cansativos e altamente suscetíveis a erros. Além disso, no contexto da indústria 4.0, o uso de tecnologias para melhorar a avaliação destes produtos torna-se cada vez mais essencial. Portanto, este trabalho tem como objetivo o aprendizado de descritores e de classificadores de padrões mais adequados para a classificação automática de produtos e o controle de qualidade em indústrias do ramo alimentício, mais especificamente envolvendo biscoitos. Para tanto, uma avaliação experimental extensiva foi realizada considerando diferentes abordagens de aprendizado (tradicionais e baseadas em redes neurais convolucionais). A partir dos resultados obtidos, é possível observar que a metodologia proposta pode proporcionar um controle de qualidade mais efetivo para a empresa, atingindo acurácias de até 99%. Pode-se evitar o oferecimento de produtos em não conformidade aos padrões de qualidade no mercado, melhorando a credibilidade da marca junto ao consumidor, sua rentabilidade e consequentemente sua competitividade.

Palavras-chave: Aprendizado de Máquina; Processamento de imagens; Classificação; Aprendizado Profundo; Controle de Qualidade; Visão Computacional.

ABSTRACT

SILVA, Mardlla S. LEARNING STRATEGIES TOWARDS IMPROVEMENTS IN CLASSIFICATION AND QUALITY CONTROL PROCESSES FOR THE FOOD INDUSTRY. 62 f. Dissertação – Mestrado – Programa de Pós-Graduação em Informática, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2020.

Considering the great competition among industries, one of the main factors that make companies market leaders is the quality of their products. However, the techniques applied to quality control are often faulty or inefficient, due to the great dependence on the human factor, which makes the applied procedures tiring and highly susceptible to errors. Moreover, in the context of industry 4.0, the use of technologies to improve the evaluation of these products becomes increasingly essential. Therefore, this work aims to learn the most appropriate descriptors and pattern classifiers for automatic product classification and quality control in food industries, more specifically regarding cookies. For this, an extensive experimental evaluation was performed, considering different learning approaches (traditional and based on convolutional neural networks). From the obtained results, it is possible to observe that the proposed methodology can provide a more effective quality control for the company, reaching accuracies of up to 99%. It is possible to avoid offering products that do not comply with the quality standards in the market, improving the credibility of the brand with the consumer, its profitability and consequently its competitiveness.

Keywords: Machine Learning; Image Processing; Classification; Deep Learning; Quality Control; Computer Vision.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS

ACC	*Auto Color Correlogram*
BIC	*Border/Interior Pixel Classification*
CAM	*Class Activation Mapping*
CEDD	*Cor and Edge Directivity Descriptor*
CNNs	*Convolutional Neural Networks*
FCTH	*Fuzzy cor and Textura Histogram*
GAP	*Global Average Pooling*
GCH	*Global cor Histogram*
Grad-CAM	*Gradient-weighted Class Activation Mapping*
IoTs	*Internet of Things*
JCD	*Joint Composite Descriptor*
k-NN	*k-Nearest Neighbor*
LBP	*Local Binary Pattern*
LCH	*Local cor Histogram*
MLP	*Multilayer Perceptron*
NB	*Naive Bayes*
PHOG	*Pirâmide de Histogramas de Gradientes de Orientação*
RCS	*Reference Color Similarity*
RF	*Random Forest*
RNA	*Redes Neurais Artificiais*
SVM	*Support Vector Machines*
TI	*Tecnologia da Informação*

SUMÁRIO

1 INTRODUÇÃO

Segundo Nashat et al. (2011) o setor de panificação pode ser considerado um dos mais importantes da indústria alimentar. O setor de panificação engloba a produção de biscoitos, massas e pães sendo que em 2018 no Brasil foram vendidas mais de 1 bilhão de toneladas de biscoitos (HELMAN, 2019). No entanto, para manter o setor em constante crescimento, a inovação tecnológica é essencial no processo produtivo.

Isto é comprovado com a quarta onda de avanços tecnológicos industriais, conhecida como indústria 4.0 (RUBMANN et al., 2015), em que sensores, maquinário e sistemas de *Tecnologia da Informação* - TI são interconectados ao longo de toda cadeia produtiva, criando um sistema ciberfisico. Nesse contexto, dados devem ser coletados e analisados para a previsão de falhas, tornando o processo produtivo mais robusto e eficiente, aumentando assim a produtividade e a qualidade dos produtos.

Para o setor alimentício um dos fatores mais importantes é a qualidade do que se é produzido. O fator qualidade está intrinsecamente ligado à satisfação do consumidor, que é peça chave para o sucesso ou não de uma empresa. Qualquer deficiência no produto pode gerar insatisfação (aborrecimento, reclamações, reivindicações) do cliente (JURAN, 1974). Consumidores insatisfeitos pela qualidade de determinados produtos podem gerar prejuízos e comprometer a competitividade da empresa.

A análise visual de qualidade realizada por funcionários é ineficiente e inviável de ser realizada, considerando que uma grande quantidade de produtos são produzidos diariamente em uma indústria. Proporcionar produtos com maior qualidade pode requerer determinados investimentos tecnológicos e, consequentemente, demandar aumento de custos.

Nesse sentido, deve-se levar em consideração o desempenho (relacionado ao aumento de produtividade e de eficiência) em empresas resultante da implementação de novas tecnologias. Nesse processo de inovação tecnológica, para permanência em um mercado cada vez mais competitivo, a qualidade do produto não pode ser perdida (MARTINS; LAUGENI, 2005). Sendo assim, é fundamental a utilização de técnicas e ferramentas de aprendizado de máquina

mais adequadas considerando não apenas eficácia, mas também eficiência.

1.1 JUSTIFICATIVA

Com este novo paradigma da industria 4.0, que visa aproveitar ao máximo a tecnologia das máquinas por meio de *Internet of Things* - IoTs, há um grande potencial para a melhoria no processo de fabricação (WANG et al., 2018) e consequentemente na qualidade do produto. A partir do processo produtivo cada vez mais automatizado, torna-se cada vez mais importante um processo mais eficiente de verificação global de produtos em não conformidade aos padrões de qualidade, de forma a acompanhar os avanços e a velocidade de produção em larga escala.

Nesse sentido é de extrema importância a automatização do processo de identificação dos produtos que fogem dos padrões estabelecidos pela empresa. Nestes casos o uso de técnicas de visão computacional tem apresentado resultados significativos para classificação dos produtos (SIVAKUMAR; SRILATHA, 2016; NASHAT et al., 2014; SRIVASTAVA et al., 2014; DAVIDSON et al., 2001; BROSNAN; SUN, 2004; LU, 2016).

Apesar de alguns esforços envolvendo desde à aquisição, processamento e classificação de imagens na indústria de alimentos (SIVAKUMAR; SRILATHA, 2016; NASHAT et al., 2014; SRIVASTAVA et al., 2014; DAVIDSON et al., 2001; BROSNAN; SUN, 2004; LU, 2016), não existem trabalhos na literatura que englobem diferentes categorias de biscoitos e diferentes tipos de danos em cada categoria. Além disso, é importante explorar técnicas de aprendizado considerando eficácia e eficiência (i.e. acurácia nas classificações, bem como tempos e recursos computacionais).

1.2 OBJETIVOS

1.2.1 OBJETIVO GERAL

Este projeto tem como objetivo o aprendizado de descritores e de classificadores de padrões, de forma a melhorar a classificação e o controle de qualidade de biscoitos em uma indústria do ramo alimentício.

1.2.2 OBJETIVOS ESPECÍFICOS

Para atingir o objetivo geral definido na seção 1.2.1, foram estabelecidos os seguintes objetivos específicos:

- Aquisição e preparação dos conjuntos de imagens, considerando diferentes categorias de biscoitos, bem como diferentes tipos de danos;
- Extração de características (baseadas em cor, textura e forma) obtidas por meio de descritores tradicionais;
- Extração de características profundas por meio de arquiteturas de redes neurais convolucionais;
- Análise de desempenho de cada descritor utilizando classificadores tradicionais e arquiteturas de redes neurais convolucionais;
- Comparação entre as abordagens de aprendizado supervisionado tradicional e de aprendizado *end-to-end* por meio das arquiteturas de redes neurais convolucionais

1.3 ORGANIZAÇÃO DO TEXTO

O presente trabalho apresenta a seguinte organização:

- No capítulo 2 são introduzidos conceitos referentes ao aprendizado de máquina e às principais técnicas e trabalhos relacionados aplicados na área de visão computacional.
- No Capítulo 3 é apresentada a metodologia de aprendizado de descritores e de classificadores proposta.
- No Capítulo 4 são apresentados os experimentos realizados, incluindo a descrição do conjunto de imagens e cenários, e discussão dos resultados obtidos.
- No Capítulo 5 são expostas as considerações finais.

2 FUNDAMENTAÇÃO TEÓRICA

O desenvolvimento de sistemas de visão computacional para soluções de problemas de classificação envolvendo diferentes domínios de aplicação (inclusive no setor de alimentos) vem crescendo exponencialmente (JACKMAN; SUN, 2013; BROSNAN; SUN, 2004). No entanto, há muito o que ser explorado nesta área, principalmente em relação ao aprimoramento das técnicas e ferramentas de aprendizado de máquina já existentes.

A Figura 1 apresenta as principais etapas de um sistema de visão computacional. A primeira etapa diz respeito à aquisição das imagens. Após essa etapa, técnicas de pré-processamento podem ser necessárias para tratamento (como eliminação de ruídos e segmentação) das imagens (GONZALEZ; WOODS, 2001). A segmentação consiste em separar os objetos de interesse da imagem. Tais objetos devem então ser representados por um conjunto de atributos (características) que melhor os descrevem (*aprendizado de descritores*). Em seguida, modelos podem ser treinados (*aprendizado de classificadores*) e aplicados para classificação.

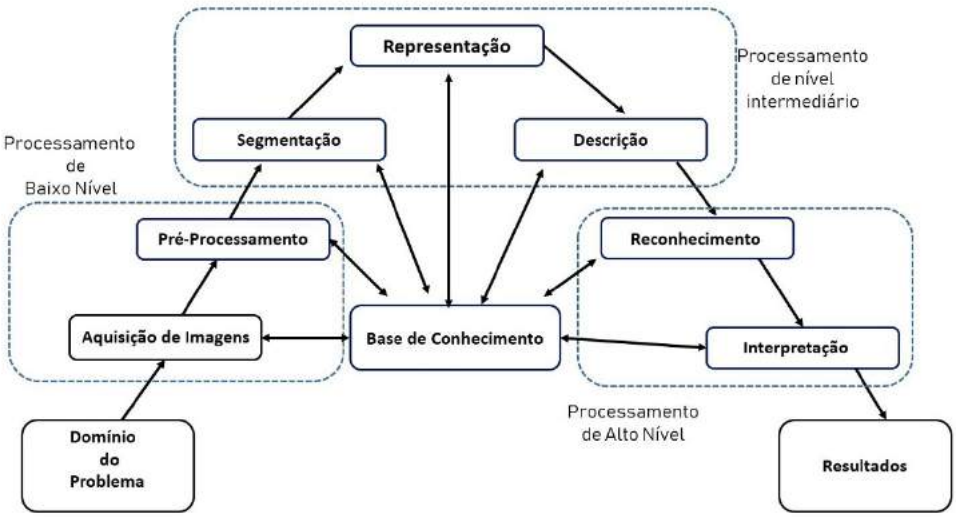


Figura 1: Principais etapas de um sistema de visão computacional.

Fonte: Autória Propria

2.1 APRENDIZADO DE DESCRITORES

Idealmente os descritores de imagem deveriam realizar a extração de características relevantes de uma determinada imagem, de forma similar a um observador humano. No entanto, apesar de alguns avanços, há ainda muito a ser explorado e melhorado, dado o conhecimento atual sobre visão, cognição e emoção humana. Entre as características mais utilizadas para descrever uma imagem estão aquelas definidas como primitivas (baixo nível), derivadas de três elementos fundamentais da imagem: distribuição de intensidades (cores), textura e forma.

As características baseadas em cores são amplamente utilizadas em numerosas aplicações, devido ao baixo custo computacional e invariância nas operações, tais como rotações e translações nas imagens. A descrição de uma dada imagem relacionada à distribuição global de cor é normalmente realizada pela construção de um histograma de cor. Em um histograma são computadas as quantidades de *pixels* da imagem para cada intensidade de cor. Apesar de apresentar custo linear em relação à quantidade de *pixels* da imagem, os histogramas apresentam capacidade de discriminação reduzida e não fornecem informações sobre a distribuição espacial das cores em uma imagem. Imagens distintas, embora visivelmente diferentes, podem apresentar histogramas idênticos.

Diversas propostas têm sido apresentadas para tratar tal problema, incluindo a combinação de características de cor e textura. Diferentemente da cor, a textura ocorre sobre uma determinada região ao invés de um ponto (*pixel*). Além disso, dado que apresenta certa periodicidade e escala, pode ser descrita em termos de direção, rugosidade, contraste, entre outros.

As características baseadas em forma, embora em geral envolvam processos não triviais (o que pode gerar um custo computacional mais elevado), também são interessantes quando utilizadas em alguns domínios de aplicação. É importante que tais características sejam invariantes às transformações geométricas, tais como translação, rotação e escala. Inúmeros métodos têm sido desenvolvidos, incluindo a combinação e seleção de características relevantes para caracterização de imagens.

Os processos de descrição de imagens (i.e. extração de características) tradicionais mencionados anteriormente são intrinsecamente relacionados ao contexto do problema. Portanto, tais características podem apresentar problemas em relação à generalização.

Sendo assim, atualmente métodos de descrição de imagens baseados em arquiteturas de aprendizado profundo, como as redes neurais convolucionais (*Convolutional Neural Networks* - CNNs) (MOSAVI, 2017) têm sido amplamente utilizados. Este tipo de arquitetura é

capaz de aprender as características relacionadas a um problema por meio de uma representação hierárquica das características desde as de baixo nível até as de alto nível.

2.2 APRENDIZADO DE CLASSIFICADORES

Classificadores de padrões podem ser treinados a partir de exemplos e estratégias de aprendizado de máquina. Existem diferentes abordagens de aprendizado.

No aprendizado supervisionado, o modelo de aprendizado é obtido por meio de um conjunto de treinamento de dados previamente rotulado. Na literatura, inúmeras propostas de algoritmos supervisionados podem ser encontradas, dentre elas: k-Nearest Neighbor - k-NN (RANI; VASHISHTHA, 2017), árvores de decisão - J48 (ARWAN, 2018), Random Forest - RF (BREIMAN, 2001), Support Vector Machines - SVM (HE et al., 2018), Naive Bayes - NB (BRILHADOR, 2014). Considerando que são amplamente utilizados e apresentam bom desempenho em diversos domínios de aplicação, tais classificadores são explorados no presente trabalho.

Redes Neurais Convolucionais (*Convolutional Neural Networks* - CNNs) também devem ser investigadas para a obtenção das características profundas, bem como para o processo de aprendizado *end-to-end* no processo de aprendizado profundo.

2.3 APRENDIZADO PROFUNDO

Técnicas de aprendizado profundo (*deep learning*) têm sido exploradas para prover melhorias e soluções a vários problemas de visão computacional. Utilizam múltiplas camadas para extrair progressivamente características de alto nível a partir dos dados. Por exemplo, em processamento de imagens, camadas inferiores podem identificar bordas, enquanto camadas superiores podem identificar itens mais significativos, como dígitos, letras ou faces.

Modelos de aprendizado profundo, especificamente CNNs, são baseados em Redes Neurais Artificiais - RNA. Uma RNA, inspirada pelas redes neurais biológicas que constituem o cérebro humano, é baseada em uma coleção de unidades conectadas denominadas neurônios artificiais (*perceptrons*). Cada conexão (sinapse) entre neurônios pode transmitir um sinal para outro neurônio. A Figura 2 mostra um exemplo do funcionamento básico da rede *perceptron*. O funcionamento básico ocorre da seguinte forma:

- Os sinais de entrada x_1, x_2, x_3, x_n são ponderados/multiplicados pelos pesos y_1, y_2, y_3, y_n.

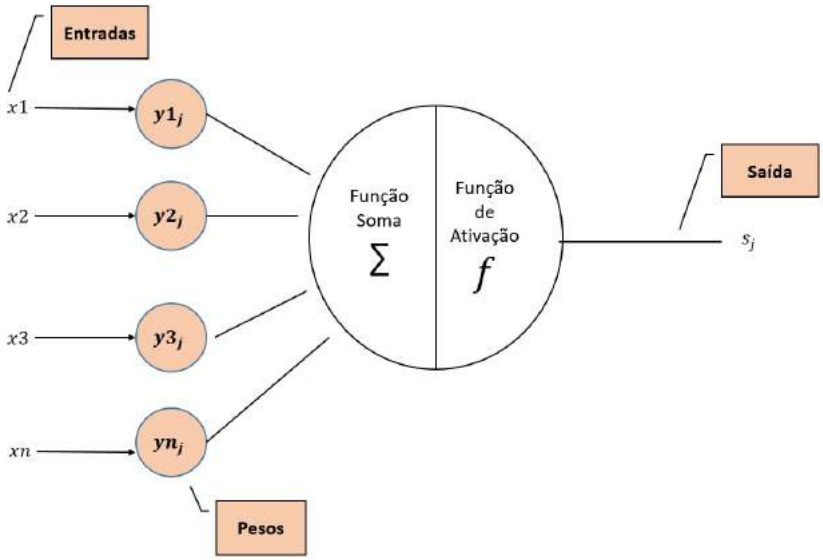


Figura 2: Exemplo de uma rede perceptron.
Fonte: Adaptado de (ALVAREZ et al., 2017)

- A função soma Σ recebe todos os sinais e realiza a soma ponderada dos sinais.
- Ao resultado, é somado o limiar de ativação $\int$ (também chamado de bias ou parâmetro polarizador), soma essa conhecida como potencial de ativação; o bias é uma constante que serve para aumentar ou diminuir a entrada, de forma a transladar a função de ativação.
- Como nas redes neurais biológicas, a saída da rede é alimentada em outros Perceptrons..

A CNN é uma rede múltiplas camadas, que utiliza a convolução em pelos menos uma de suas camadas conforme demonstrado no exemplo da Figura 3. O diferencial das CNNs está nas diversas camadas convolucionais, que aplica uma função matemática de convolução nos dados de entrada e depois realiza o agrupamento (*pooling*). A saída da convolução é passada para a próxima camada convolucional até chegar na última camada conhecida como camada densa que normalmente é representada por uma rede *perceptron* de múltiplas camadas (do inglês *Multilayer Perceptron* - MLP) (VARGAS et al., 2016). Essa arquitetura foi inspirada no processo visual dos mamíferos, o qual sugere que os mesmos percebem visualmente o mundo ao seu redor de modo hierárquico, por meio de camadas de *clusters* dos neurônios. Assim, *clusters* são ativados hierarquicamente, e cada um detecta um conjunto de atributos sobre o que foi visto.

Na Figura 4 tem-se um exemplo de um gráfico de uma rede neural *perceptron*, com dados de entrada (tensor X) multiplicados pelos pesos (tensor W) e somados pelo limiar de ativação (tensor b). O tensor de custo de saída (C) obtido da função de ativação ReLU.

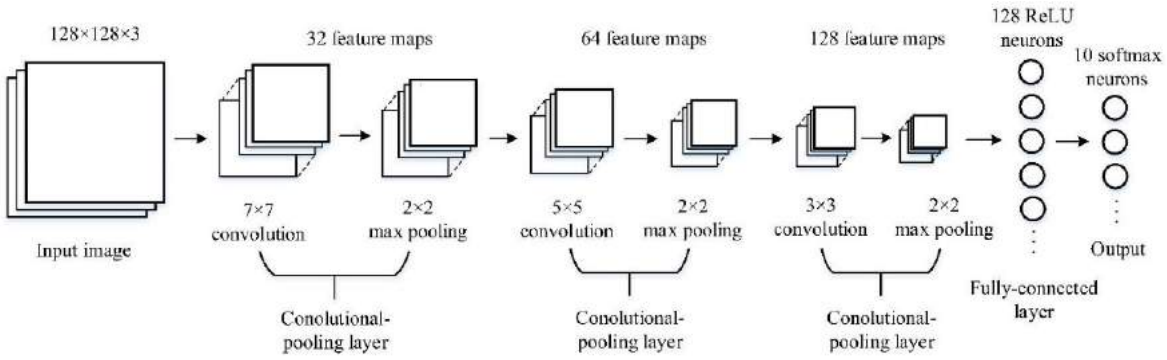


Figura 3: Exemplo de topologia da Rede Neural Convolucional.

Fonte: (LU, 2016)

A função ReLU é a unidade linear retificada. É definida como:

f(x) = max (0, x) a saída é o valor de x ou 0 (zero), depende de quem for maior.

Exemplos: se x = -1, então f(x) = 0(zero); se x = 0.7, então f(x) = 0.7.

ReLU é a função de ativação mais amplamente utilizada ao projetar redes neurais atualmente. Primeiramente, a função ReLU é não linear, o que significa que pode-se facilmente copiar os erros para trás e ter várias camadas de neurônios ativados pela função ReLU (GOODFELLOW et al., 2016).

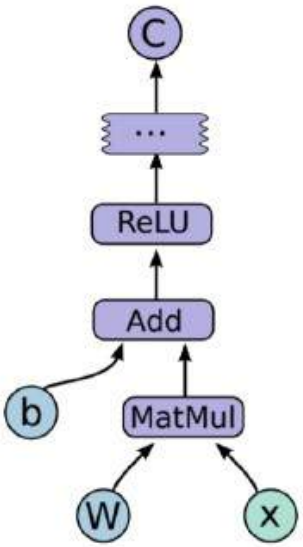


Figura 4: Exemplo de fluxo de dados gráficos da função ReLU.

Fonte: (ABADI et al., 2016)

2.4 MÉTRICAS DE VALIDAÇÃO

Uma das principais tarefas na construção de qualquer modelo de aprendizado de máquina é avaliar o seu desempenho. Existem diversas métricas de validação para mensurar a qualidade do modelo. No presente projeto serão consideradas métricas como: acurácias médias, precisão, revocação, F1-Score, tempos de treinamento e de teste e matriz de confusão (SAMMUT; WEBB, 2010).

Além disso, a técnica GRAD-CAM também deve ser considerada no presente projeto, dado que permite interpretar modelos de redes neurais convolucionais de uma forma diferente das métricas tradicionais (como acurácia, precisão, revocação, dentre outras). Nesse caso, trata-se de uma forma visual de verificar onde as ativações de uma determinada camada do modelo estão concentradas - usando os gradientes da última camada de ativação para a classe desejada - fornecendo destaque aos padrões mais importantes de uma dada imagem e facilitando, dessa forma, a verificação do funcionamento esperado da rede (SELVARAJU et al., 2019).

O método Grad-CAM é uma generalização do *Class Activation Mapping* - CAM, o qual utiliza a saída da última camada convolutiva (imediatamente antes da camada média de *pooling*), juntamente com as previsões, para fornecer uma visualização do mapa de calor da razão pela qual o modelo tomou a sua decisão. Nesse sentido, torna-se uma ferramenta bastante útil para a interpretação dos resultados obtidos. Mais precisamente, em cada posição da camada convolutiva final, tem-se vários filtros como na última camada linear. Portanto, é possível calcular o produto de pontos dessas ativações com os pesos finais para obter, para cada localização no mapa de características, a pontuação da característica que foi utilizada para tomar uma decisão (ZHOU et al., 2015).

Basicamente a diferença entre os dois metódos está em suas arquiteturas, o Grad-CAM não requer uma arquitetura CNN específica, já o CAM requer uma arquitetura que aplique o *Global Average Pooling* - GAP aos mapas de características convolucionais finais, seguida de uma única camada totalmente conectada que produz as previsões. Deste modo, Grad-CAM é uma forma mais adequada de generalização para este tipo de análise, visto que permite tanto ser utilizada em diversas arquiteturas como também sem depender de uma camada em específico (SELVARAJU et al., 2016).

2.5 TRABALHOS RELACIONADOS

De acordo com Nashat et al. (2014), um dos principais problemas no trabalho de classificação de biscoitos em tempo real, refere-se à elevada quantidade de dados e à necessidade de tempos computacionais otimizados, os quais requerem técnicas de visão computacional adequadas. Apesar de haver na literatura muitos trabalhos relacionados à inspeção de qualidade de alimentos (LEMANZYK et al., 2015; CUBEDDU et al., 2014; SIVAKUMAR; SRILATHA, 2016), poucos podem ser utilizados na classificação de produtos devido às restrições computacionais e de tempo mencionado acima.

Em seu trabalho, Nashat et al. (2011) explorou apenas um tipo de dano em um tipo de biscoito específico. Neste caso, o dano trabalhado foi a detecção automática de rachaduras no tipo de biscoito *Crack*, este é o único dano que geralmente é analisado neste tipo de biscoito. Diferentemente de trabalhos como Nashat et al. (2014) e Nashat et al. (2011) que exploraram um único tipo de biscoito e apenas um defeito específico, o presente projeto visa à classificação de tipos diferentes de biscoitos, além de analisar a conformidade ou não conformidade aos padrões de qualidade de acordo com a empresa.

Nos estudos relatados por Srivastava et al. (2014) foram considerados dois tipos de biscoitos (*cracker* e *cookies*). Para o biscoito *cracker* foram observadas questões de falhas, fissuras, bem como referentes a condições ambientais relacionadas à umidade do biscoito, embalagem (aberta ou bem fechada). No entanto para o biscoito do tipo *cookie* foi realizada a análise e a contagem da quantidade de pedaços de chocolate no biscoito. Apesar de considerarem mais de um tipo de biscoito, para ambos os tipos, os autores exploram apenas técnicas de visão computacional tradicionais. Como já bem conhecido na literatura (bem como mencionado pelos autores), tais técnicas são treinadas especificamente para um dado tipo de biscoito e tipo de dano. Nenhuma técnica consegue apresentar um desempenho (em termos de eficácia e eficiência) adequado para todos os tipos de biscoitos e danos. É necessário um estudo e avaliação experimental extensiva para obtenção das melhores técnicas para descrição e classificação. Portanto, o presente trabalho explora diferentes abordagens de aprendizado (tradicionais e baseadas em redes neurais convolucionais), de forma a obter as técnicas mais adequadas para a classificação automática de diferentes tipos de biscoitos e o controle de qualidade em indústrias do ramo alimentício.

3 METODOLOGIA

Pretende-se com este trabalho obter as melhores abordagens de aprendizado de descritores e de classificadores, para que as mesmas sejam aplicadas em uma indústria de alimentos voltada à produção de biscoitos. Para tanto, diferentes extratores e classificadores devem ser analisados tanto para i-) identificação de biscoitos correspondentes (ou não) aos padrões exigidos pela empresa (i.e. classificação binária), como ii-) a identificação de diferentes tipos de biscoitos (i.e. classificação multiclasse). A Figura 5 apresenta o pipeline da metodologia de desenvolvimento da abordagem proposta (SILVA et al., 2020).

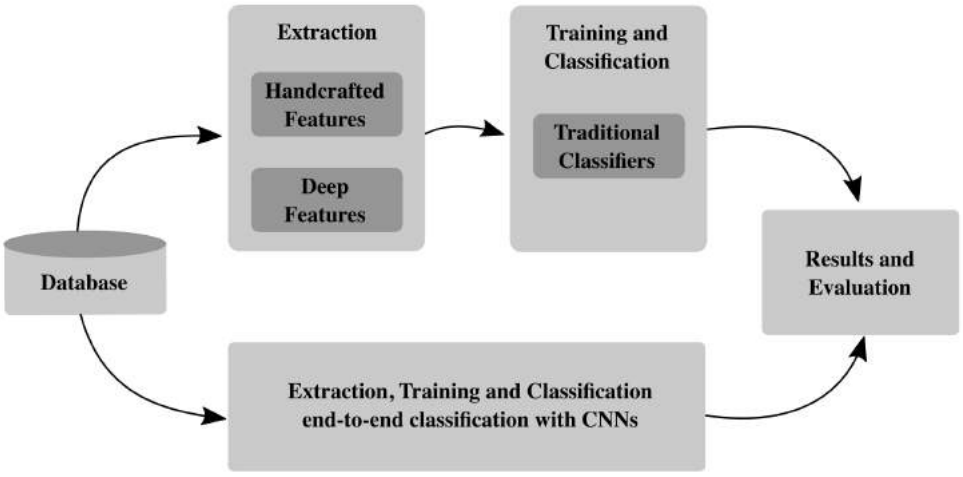


Figura 5: Pipeline da metodologia proposta.

Inicialmente, foram obtidas as imagens a partir de uma linha de produção de uma indústria. Após a organização dos conjuntos de dados (ver descrição na Seção 4.1), há dois fluxos principais. A partir do primeiro fluxo, é possível realizar a extração de características (tradicionais e profundas), utilizando estratégias tradicionais e as arquiteturas CNNs, respectivamente. Considerando as características profundas, as mesmas podem ser obtidas de uma determinada camada (geralmente a última camada densa) ou refinando o modelo CNN pré-treinado para trabalhar com um novo domínio. Diferentes arquiteturas CNN podem ser utilizadas. Neste caso, foi aplicada a técnica de aprendizado por transferência (*transfer learning*), em que um modelo, treinado em uma tarefa, é reaproveitado em uma segunda tarefa relacio-

nada. Os modelos pré-treinados (GOODFELLOW et al., 2016) geralmente foram treinados em um conjunto de dados maior, a partir de um domínio geral (ou seja, não relacionado aos biscoitos). Foram considerados os pesos aprendidos (ou seja, parâmetros treináveis) a partir das CNNs como características profundas. Portanto, estas características (tradicionais e profundas) extraídas podem ser avaliadas pelos classificadores tradicionais.

Em relação ao segundo fluxo, na etapa de refinamento, é avaliado o processo de aprendizado e de classificação *end-to-end*. Neste caso, a camada totalmente conectada é retreinada e a camada de classificação final (por exemplo, *softmax*) é substituída, de forma a produzir o número correto de probabilidades, de acordo com o conjunto de dados considerado.

Posteriormente, a metodologia proposta permite realizar análises entre diferentes tipos de extratores e de classificadores, e avaliar a configuração mais apropriada (par extrator/classificador) para a classificação dos biscoitos. O Algoritmo 1 apresenta detalhes da metodologia proposta.

Algoritmo 1: Metodologia proposta

input : conjunto de dados de imagens $\mathscr{D}$
output : melhor modelo de aprendizado M^{Ω}
auxiliaries: E: conjunto de extratores de características; H: conjunto de arquiteturas CNN pré-treinadas; Feats: conjuntos de características tradicionais e profundas; TrainSets and TestSets: conjuntos de treinamento e de testes; perTrain and perTest: percentagens do conjunto de treinamento e teste; nsplits: número de divisões do conjunto; $\mathscr{C}$: conjunto de classificadores tradicionais; ModelSets: conjuntos de modelos de aprendizado; AccSets: acurácias médias; TrainSetIds and TestSetIds: identificadores das amostras de treinamento e de teste de a partir de cada divisão do conjunto, $MaxAcc^{\Omega}$: acurácia máxima.

```
HandCraftedFeatures ← getHCFeatures(𝒟, E);
DeepFeatures ← getDeepFeatures(𝒟, H);
Feats ← HandCraftedFeatures ⋃ DeepFeatures;
for each i ∈ Feats_i, i = 1,...,nf do
    TrainSets_i, TestSets_i ← stratifiedSplits(Feats_i, perTrain, perTest, nsplits);
    for each j ∈ 𝒞_j do
        ModelSets_ij ← generateModels(TrainSets_i, 𝒞_j);
        AccSets_ij ← testModels(TestSets_i, ModelSets_ij);
    end
end
for each i ∈ H_i, i = 1,...,nh do
    AccSets ← AccSets ⋃ end2end(TrainSetIds, TestSetIds, H_i);
end
MaxAcc^Ω ← findMaxAcc(AccSets);
```

4 EXPERIMENTOS

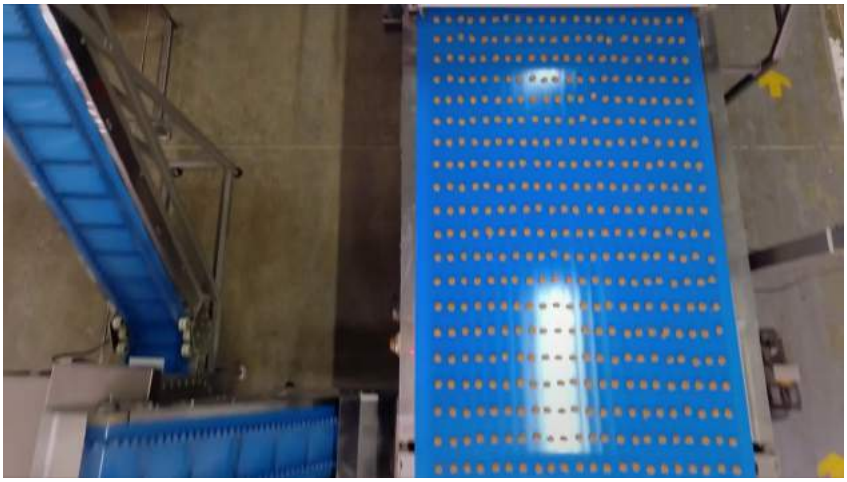
Neste Capitulo são apresentados os experimentos realizados para validação da abordagem proposta. Para tanto, são apresentadas as descrições dos conjuntos de dados (Seção 4.1) e dos cenários (Seção 4.2), bem como as discussões dos resultados obtidos (Seção 4.3).

4.1 DESCRIÇÃO DO CONJUNTO DE DADOS

A etapa inicial deste trabalho consiste na construção do conjunto de dados das imagens de biscoitos. Foram necessários funcionários da linha de produção para recolher os biscoitos, já que não era possível visitar frequentemente a linha de produção por pessoas não autorizadas. Os funcionários recolhiam as amostras uma vez por semana, pois os biscoitos são produzidos sob demanda. Assim, um determinado tipo de biscoito não é produzido todos os dias. A aquisição das imagens foi realizada utilizando uma câmera de 8 megapixels com uma resolução de 3264x2448 *pixels* Full HD (1920x1080 *pixels*) com 60 fps. Tais imagens foram obtidas em iluminação ambiente, sempre no período da tarde, com um ângulo reto (90^o graus) e com uma distância de aproximadamente 30 cm entre a câmera do celular e a amostra. A Figura 6 apresenta imagens da linha de produção dos biscoitos na fábrica.

Foram selecionados diferentes tipos de biscoitos apresentando qualidade padrão e não-padrão de acordo com a política da fábrica. O conjunto de dados foi dividido em 10 classes, referentes a cada sabor de biscoito e seu nível de qualidade (padrão ou não-padrão). Foram coletadas 1.000 amostras para cada classe, compondo um conjunto de 10.000 amostras para a avaliação experimental. A Tabela 1 apresenta cada classe de imagem, suas respectivas descrição e quantidade de amostras. A Figura 7 mostra exemplos de imagens das classes (cada tipo de biscoito e o seu nível de qualidade - padrão e não-padrão) do conjunto de dados. As imagens foram obtidas a partir de um fundo azul com diferentes tons, para simular a cor das esteiras após a passagem pelo forno, onde acontece a assadura ou a adição de recheio.

Como a análise de qualidade da empresa é realizada de forma visual, é utilizado para classificação um modelo de biscoito padrão, demonstrado nas imagens superiores da Figura



(a)

(b)

(c)

Figura 6: Imagens da linha de produção dos biscoitos na fábrica. (a) esteira com biscoitos. (b) produção de biscoitos do tipo cookies. (c) máquina recheadora de biscoitos do tipo tortinhas.

Fonte: Autoria Própria

7. Este modelo fica exposto ao lado da linha de produção para servir de referência para os funcionários que irão coletar os biscoitos que não seguem o padrão mínimo do modelo. Existe uma tolerância para que os biscoitos sejam classificados como não padrão. Por exemplo, para uma dada amostra chocolate (parte superior da Figura 7), o recheio transpassa um pouco a borda, porém não é classificada como um problema. No caso deste tipo de biscoitos a falta de recheio é o principal problema. Além da análise do recheio, outros padrões são verificados como peso (biscoitos cru e assado), espessura, comprimento e umidade em ambos tipos de biscoitos. Por exemplo, o comprimento máximo do biscoito tortinha é de $42,5$mm e o mínimo $41,5$mm, já o biscoito do tipo Cookie o seu comprimento máximo é de $57,0$mm e o mínimo de $50,0$mm.

Tabela 1: Descrição do conjunto de biscoito.

Classes	Descrição	Total
C_1	Vanilla - padrão	1000
C_2	Vanilla - não padrão	1000
C_3	Chocolate - padrão	1000
C_4	Chocolate - não padrão	1000
C_5	Candy - padrão	1000
C_6	Candy - não padrão	1000
C_7	Strawberry - padrão	1000
C_8	Strawberry - não padrão	1000
C_9	Cookie - padrão	1000
C_{10}	Cookie - não padrão	1000

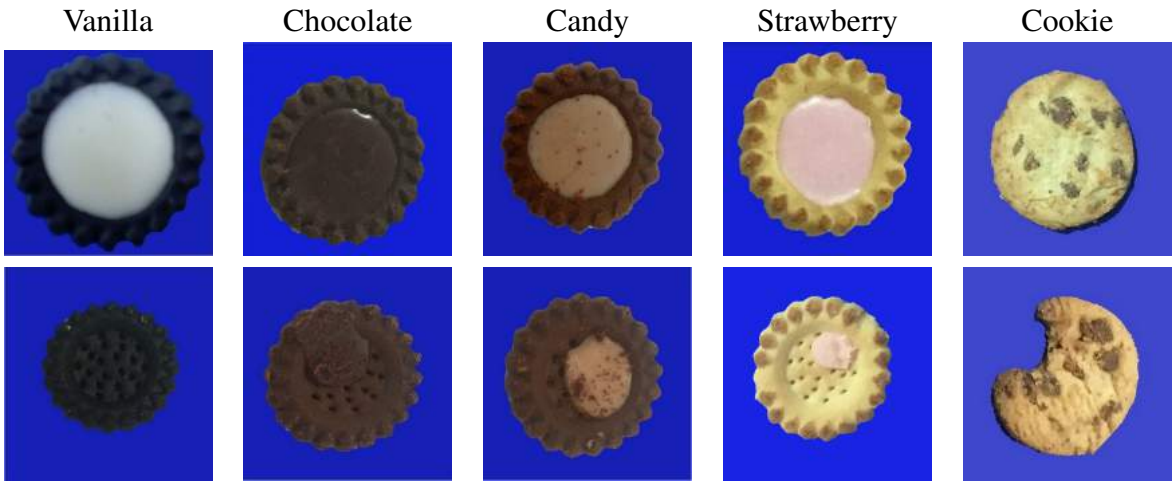


Figura 7: Exemplos de imagens de cada classe. Imagens superiores correspondem às imagens com nível de qualidade padrão e as imagens inferiores referem-se às imagens com nível de qualidade não padrão.

Fonte: Autoria Própria

4.2 DESCRIÇÃO DOS CENÁRIOS

Para realizar os experimentos as características tradicionais foram extraídas usando descritores de imagem tradicionais e as características profundas a partir de quatro arquiteturas CNNs diferentes, denominadas DenseNet161 (HUANG et al., 2016), EfficientNetB3 (TAN; LE, 2019), ResNet50 (MAHMOOD et al., 2020) e VGG19 (SIMONYAN; ZISSERMAN, 2014). Foi aplicada a técnica de aprendizado por transferência para todas as arquiteturas (pré-treinadas no conjunto de dados ImageNet (SZEGEDY et al., 2015)). Isto proporciona uma redução considerável de dados e custos de treinamento. Todas as arquiteturas foram utilizadas não apenas para extrair as características profundas a partir do conjunto de imagens, mas também para realizar o processo de aprendizado *end-to-end* (retreinando apenas as camadas densas e congelando as demais). A Tabela 2 apresenta os extratores tradicionais considerados, com seus respectivos tipos (por exemplo, cor, textura, genérico) e quantidades de características.

Tabela 2: Descritores de imagens, tipos e quantidades de características extraídas.

Descritores	Categoria	Características
ACC (MITRA et al., 2002)	cor	768
BIC (BISHOP, 2006)	cor	128
CEDD (CHATZICHRISTOFIS; BOUTALIS, 2008a)	cor	144
FCTH (CHATZICHRISTOFIS; BOUTALIS, 2008b)	cor e textura	192
Gabor (VINAYAK; JINDAL, 2017)	textura	60
GCH (WANG, 2001)	cor	255
HARALICK (LöFSTEDT et al., 2019)	textura	14
JCD (KUMAR et al., 2012)	cor e textura	336
LBP(HUANG et al., 2011)	textura	256
LCH (WANG, 2001)	cor	135
MOMENTS(VINAYAK; JINDAL, 2017)	textura	4
MPO (GUNAYDIN, 2016)	textura	6
PHOG (ZHANG; SHA, 2013)	textura	40
RCS(WANG et al., 2014)	cor	77
TAMURA (KARMAKAR et al., 2017)	textura	18
DenseNet161 (HUANG et al., 2016)	generica	4416
EfficientNetB3 (TAN; LE, 2019)	generica	3072
ResNet50 (MAHMOOD et al., 2020)	generica	4096
VGG19 (SIMONYAN; ZISSERMAN, 2014)	generica	1024

Fonte: Autoria própria.

Além disso, foram utilizados diferentes classificadores tradicionais supervisionados considerando cada tipo de característica (tradicional e profunda), e comparados com o processo de aprendizado *end-to-end* por meio do aprendizado por transferência. Para o processo de avaliação, foi considerado protocolo *hold-out*. Para isso, o conjunto foi dividido em 80% para

treinamento e 20% para testes. Foram geradas 10 partições estratificadas, mantendo a proporção de amostras em cada classe. O mesmo protocolo foi considerado tanto no processo *end-to-end* quanto no processo de classificação tradicional. Em relação aos classificadores tradicionais supervisionados, foram utilizados o *k*-NN (RANI; VASHISHTHA, 2017), J48 (ARWAN, 2018), RF (BREIMAN, 2001) e SVM (HE et al., 2018), todos eles com seus parâmetros da literatura padrão.

Todas as arquiteturas - DenseNet161 (HUANG et al., 2016), EfficientNetB3 (TAN; LE, 2019), ResNet50 (MAHMOOD et al., 2020) e VGG19(SIMONYAN; ZISSERMAN, 2014) - consideraram o mesmo conjunto de hiper-parâmetros, tais como: 25 épocas de treinamento, otimizador Adam (KINGMA; BA, 2014) com um momento de 0,9 e epsilon de 1e-5, taxa de aprendizado 3e-3, seguindo a política *One Cycle* (SMITH, 2018) para variação.

A política *One Cycle* baseia-se na variação da taxa de aprendizado e do momento de uma dada CNN. Tal política permite uma convergência mais rápida dos modelos. É possível obter os mesmos resultados em menos épocas quando comparado com o treinamento realizado com estes mesmos hiper-parâmetros com um valor fixo por diversas épocas. O processo de variação da taxa de aprendizado possui dois passos de tamanhos iguais. O primeiro consiste em variar o mesmo partindo de um valor normalmente dez vezes menor que o valor escolhido e incrementando-o até que alcance a metade do valor desejado, enquanto que na segunda porção acontece o decremento deste valor até que retorne ao valor inicial dez vezes menor que o escolhido. A variação do momento ocorre de forma inversa à descrita anteriormente (SMITH, 2018).

Em relação aos experimentos tradicionais foram realizados em um máquina Intel(R) Core(TM) i5-7200U CPU 2.50GHz - 2.70GHz, Memória RAM 16,0 GB com placa Intel(R) HD Graphics 620. Para a extração das características com extratores tradicionais foi utilizado um programa em linguagem java (BRESSAN; SAITO, 2018), configurados com os parâmetros *default* da literatura. Posteriormente, o processo de classificação foi realizado com os classificadores J-48, *K*-NN, Random Forest e SVM, utilizando a biblioteca *scikit-learn*.

Os experimentos relacionados à redes neurais convolucionais incluem as etapas de preprocessamento de dados, *Data Augmentation*, treinamento das redes, extração da características e análises Grad-CAM. Os mesmos foram todos realizados utilizando o *framework* fastai (HOWARD; GUGGER, 2020). As GPUs utilizadas foram: NVIDIA GeForce GTX 1080 Ti (NVIDIA - Accelerated Data Science Call for Proposals via the GPU Grant Program) e NVIDIA GeForce RTX 2070.

4.3 RESULTADOS

A Tabela 3 apresenta os resultados obtidos por cada combinação de extratores de características e classificadores tradicionais. A partir dos resultados, é possível observar o melhor extrator de características para cada classificador (conforme valores sublinhados). De forma geral, as características obtidas pelas arquiteturas profundas (DenseNet161, EfficientNetB3 e VGG19) se destacaram em relação às obtidas pelos descritores tradicionais. As arquiteturas DenseNet161 e EfficientNetB3 foram os melhores extratores para todos os classificadores (*k*-NN, J48, RF e SVM).

Tabela 3: Acurácias médias ± desvio padrão obtidos pela abordagem de aprendizado (de descritores e de classificadores) tradicionais. Os melhores resultados (i.e. classificadores) para cada descritor são destacados em negrito. Os melhores resultados (i.e. descritores) para cada classificador são apresentados sublinhados. Os melhores resultados (i.e maiores acurácias) obtidos são apresentados com asterisco.

Descritores	*k*-NN	J48	RF	SVM
ACC	95,80±0,63	97,60±0,52	**99,10±0,32***	91,60±0,84
BIC	96,60±0,52	**97,90±0,74**	**99,10±0,57***	91,30±0,95
CEDD	95,00±0,70	**97,60±0,52**	**98,50±0,53***	92,20±0,63
FCTH	94,70±1,25	**96,20±0,42**	**96,80±0,42**	87,10±0,99
GABOR	84,20±0,71	87,90±0,57	**93,90±0,52**	81,00±0,92
GCH	96,80±0,42	97,90±0,57	**99,00±0,47***	91,20±0,42
HARALICK	92,80±0,63	**95,50±0,53**	**96,50±0,53**	73,20±0,63
JCD	95,70±0,48	97,70±0,48	**99,00±0,67***	92,50±0,53
LBP	92,00±0,82	92,50±0,53	**95,90±0,74**	87,80±0,79
LCH	96,30±0,67	96,90±0,57	**98,90±0,32***	94,20±0,42
MOMENTS	95,00±0,47	**97,10±0,57**	**97,50±0,53**	71,50±6,64
MPO	93,50±0,85	95,90±0,32	**97,00±0,47**	75,90±1,10
PHOG	84,20±0,92	87,90±0,99	**93,90±0,57**	81,00±0,67
RCS	94,10±0,57	96,50±0,53	**97,90±0,32**	78,20±0,92
TAMURA	91,60±0,52	93,90±0,74	**96,20±0,42**	82,70±0,48
DenseNet161	**98,70±0,48***	97,40±0,52	**99,30±0,48***	97,90±0,32
EfficientNetB3	**98,70±0,33***	97,20±0,79	**99,40±0,52***	97,90±0,57
ResNet50	93,80±0,63	94,20±0,42	**98,10±0,32**	90,70±0,82
VGG19	**98,80±0,42***	96,40±0,52	**99,10±0,57***	95,20±0,42

Fonte: Autoria própria.

Analisando cada extrator, é possível também obter o classificador mais adequado (valores em negrito na Tabela 3). Dentre os classificadores, o RF apresentou as melhores acurácias para todos os extratores de características considerados. Os maiores valores de acurácia representando as melhores combinações (pares extratores e classificadores) são des-

tacadas por um asterisco. As melhores combinações (pares descritor-classificador) foram ACC-RF; BIC-RF; CEDD-RF; GCH-RF; JCD-RF; LCH-RF; DenseNet161-*k*-NN; DenseNet161-RF; EfficientNetB3-*k*-NN; EfficientNetB3-RF; VGG19-*k*-NN e VGG19-RF (ver Tabela 3). Tais pares apresentam resultados equivalentes em termos de acurácia. Entretanto, analisando a dimensionalidade dos vetores de características (ver Tabela 2), pode-se observar que o melhor par seria BIC-RF. O extrator BIC permite obter acurácias elevadas (ou equivalentes) com uma quantidade de características menor em relação aos demais.

Além da acurácia e da dimensionalidade, outras métricas (precisão, revocação, F1-Score e tempos de treinamento e de teste) foram consideradas para avaliação dos melhores pares descritor-classificador (conforme apresentadas na Tabela 4). De forma geral, o par EfficientNetB3-RF apresenta os melhores desempenhos em termos de precisão, revocação, F1-Score e tempo de teste.

Tabela 4: Resultados referentes às diferentes métricas (precisão, revocação, F1-Score, tempos de treinamento e de teste (em segundos)) obtidas com a abordagem de aprendizado tradicional pelos melhores pares descritor-classificador.

Descritor-Classificador	Precisão	Revocação	F1-Score	Tempo Teste	Tempo Treinamento
ACC-RF	98,71±0,15	98,63±0,15	98,65±0,15	37,96±7,01	395,93±17,18
BIC-RF	98,98±0,05	98,91±0,05	98,93±0,05	26,67±2,25	174,74±4,45
CEDD-RF	98,38±0,07	98,32±0,07	98,34±0,07	30,75±3,46	**75,93±0,93**
GCH-RF	98,90±0,06	98,87±0,05	98,88±0,06	**22,83±0,41**	276,82±0,48
JCD-RF	98,91±0,14	98,85±0,14	98,87±0,14	**25,64±7,70**	**74,99±0,22**
LCH-RF	98,74±0,08	98,67±0,08	98,69±0,08	26,07±0,42	538,10±10,04
DenseNet161-*k*-NN	98,82±0,32	98,83±0,26	98,77±0,33	1874,25±66,74	88,35±4,09
DenseNet161-RF	99,34±0,40	98,32±0,07	98,34±0,07	30,75±3,46	81,76±2,91
EfficientNetB3-*k*-NN	98,89±0,04	98,88±0,05	98,88±0,06	**22,93±0,30**	290,21±5,45
EfficientNetB3-RF	**99,93±0,02**	**99,92±0,05**	**99,93±0,02**	**22,98±0,54**	290,26±5,38
VGG19-*k*-NN	98,86±0,58	98,94±1,44	95,87±0,49	166,04±20,47	88,35±4,09
VGG19-RF	99,15±0,03	99,09±0,35	98,87±0,14	**25,64±7,70**	85,15±4,01

Foram também realizadas comparações entre a abordagem de aprendizado tradicional (melhores resultados obtidos pelas características – tradicionais ou profundas – juntamente com os classificadores tradicionais) e a abordagem de aprendizado *end-to-end*. Em relação aos classificadores tradicionais, a melhor acurácia foi de até $99,40\% \pm 0,52$ obtida pelo extrator EfficientNetB3 com o classificador RF (ver Tabela 3). No entanto, considerando a dimensionalidade, além da acurácia, o melhor resultado foi apresentado pelo descritor BIC e classificador RF ($99,10\% \pm 0,57$). Os processos de aprendizado *end-to-end* atingiram acurácias de $98,92\% \pm 0,17$; $98,71\% \pm 0,16$; $98,72\% \pm 0,22$ e $98,63\% \pm 0,19$ pelas arquiteturas DenseNet161, EfficientNetB3, ResNet50 e VGG19, respectivamente (ver Tabela 5). Pode-se observar que ambas as abordagens de aprendizado (tradicional e *end-to-end*) apresentam resultados equi-

valentes. No entanto, apesar de apresentar determinado custo, devido à necessidade de aprendizado dos pesos para as arquiteturas, a abordagem *end-to-end* não requer o estudo dos melhores descritores e classificadores, conforme a abordagem de aprendizado tradicional.

Tabela 5: Resultados referentes às métricas (acurácia, precisão, revocação, F1-score) obtidas pela abordagem de aprendizado *end-to-end* por meio das arquiteturas CNNs (DenseNet161, EfficientNetB3, ResNet50, VGG19).

Arquiteturas CNNs	Acurácia	Precisão	Revocação	F1-Score
DenseNet161	**98,92±0,17**	**98,94±0,15**	**98,92±0,17**	**98,91±0,17**
EfficientNetB3	**98,71±0,16**	**98,75±0,15**	**98,71±0,16**	**98,71±0,16**
ResNet50	**98,72±0,22**	**98,76±0,23**	**98,72±0,24**	**98,72±0,24**
VGG19	**98,63±0,19**	**98,67±0,17**	**98,63±0,19**	**98,63±0,19**

A análise do processo de treinamento *end-to-end* das CNNs pode trazer informações interessantes para a comparação entre as arquiteturas, como por exemplo na Figura 8 em que são exibidas as acurácias ao longo das (25) épocas de treinamento. Nesta constata-se que em épocas iniciais há muita variação nas acurácias de todas as arquitetura. Uma das explicações para isto pode ser devido à variação da taxa de aprendizado durante as épocas, segundo a política *One Cycle* considerada e, conforme o valor deste hiper-parâmetro passa a diminuir a partir da metade das épocas de treinamento, as acurácias destas arquiteturas tendem a se estabilizarem em valores similares.

É possível também comparar as arquiteturas em relação ao tempo necessário para o treinamento. Na Figura 9, nota-se que a arquitetura DenseNet161 apresentou os maiores tempos de treinamento, a EfficientNetB3 e a VGG19 apresentaram tempos bastante similares e a ResNet50 apresentou o menor tempo de treinamento entre todas as arquiteturas consideradas. Sendo assim, embora as arquiteturas tenham apresentado acurácias equivalentes (ver Tabela 5 e Figura 8), a arquitetura ResNet50 apresentou um desempenho satisfatório, considerando tanto acurácia (Tabela 5) como tempo de treinamento (Figura 9).

Além das acurácias médias gerais, podem ser observadas as acurácias por classe por meio das matrizes de confusão (Figuras 10-13) para ambas as abordagens de aprendizado (tradicional e profundo). Analisando as matrizes de confusão obtidas pela abordagem de aprendizado tradicional (Figuras 10-12), é possível notar que todos os pares (descritor-classificador) apresentaram resultados significativos (i.e. uma maior concentração de valores na diagonal principal da matriz). A matriz de confusão utilizando o extrator BIC e o classificador RF (ver figura 10(b)) apresenta um dos melhores resultados (i.e. menos confusões entre as classes). No entanto, ainda há certa confusão entre algumas classes, e.g. (6) amostras da classe C_3 foram preditas

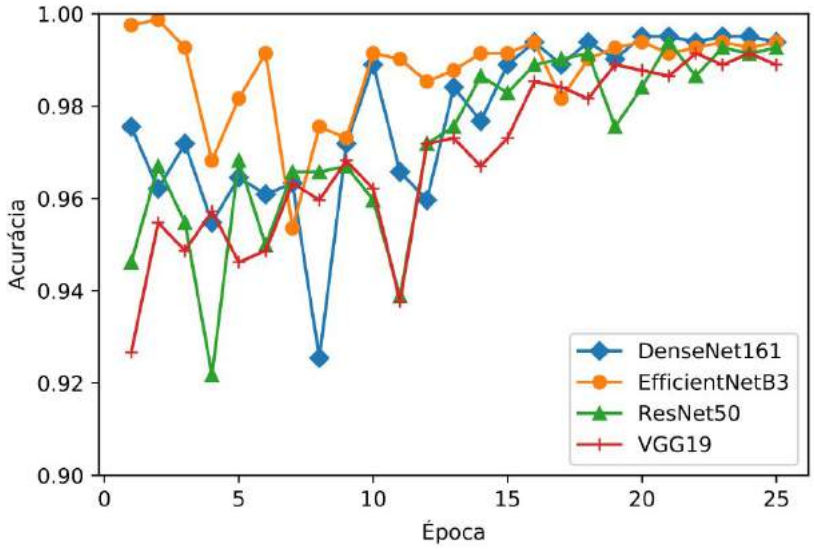


Figura 8: Representação gráfica das acurácias obtidas pelas arquiteturas DenseNet161, EfficientNetB3, ResNet50 e VGG19 durante as (25) épocas de treinamento.

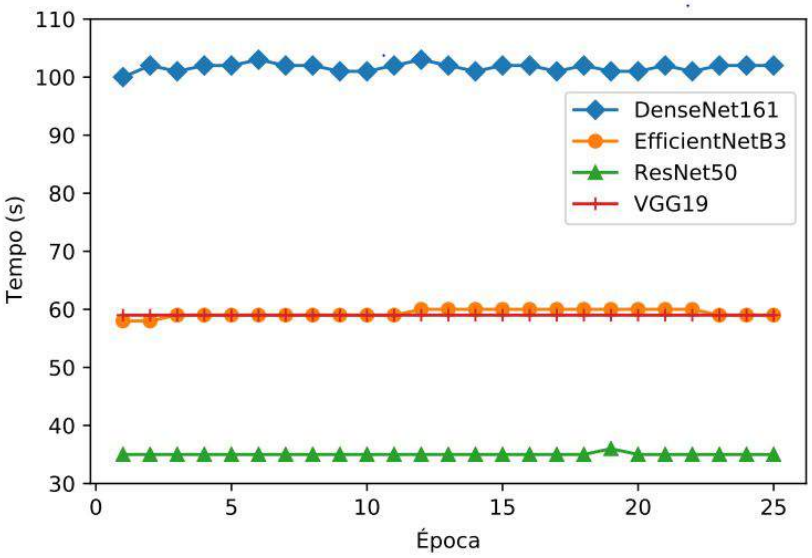


Figura 9: Representação gráfica dos tempos de treinamentos das arqtuiteturas DenseNet161, EfficientNetB3, ResNet50 e VGG19 durante as (25) épocas de treinamento.

incorretamente, i.e. a classe C_3 rotulada como classe C_8.

As principais confusões ocorreram entre as classes (no formato *trueLabel-predictedLabel*), tais como: C_8-C_7 (Strawberry - não padrão e Strawberry - padrão); C_3-C_8 (Chocolate - padrão e Strawberry - não padrão); C_6-C_5 (Candy - não padrão e Candy padrão). Tais classes foram confundidas por quase todos os (12) melhores pares (descritor-classificador), sendo em $10/12$; $7/12$; e $5/12$, respectivamente. Os piores resultados (maiores confusões) foram obtidos por

meio do VGG19-*k*-NN (Figura 12(c)) entre as classes C_8-C_7 (em 22 amostras) e C_6-C_5 (em 15 amostras).

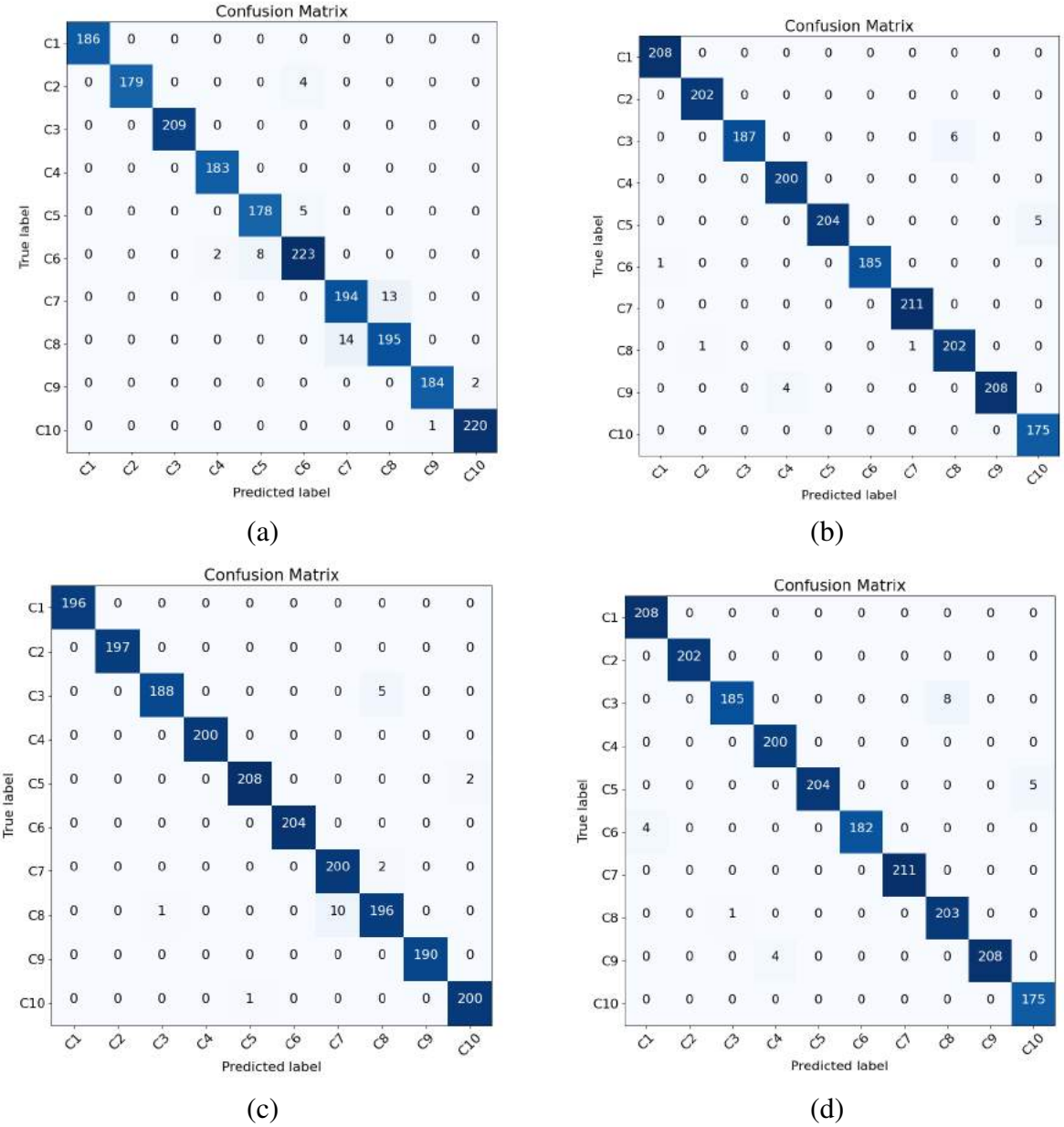


Figura 10: Matrizes de confusão obtidas pela abordagem de aprendizado tradicional utilizando as melhores combinações (pares descritor-classificador): (a) ACC-RF. (b) BIC-RF. (c) CEDD-RF. (d) GCH-RF.

Comportamentos similares podem ser observados a partir das matrizes de confusão referentes à abordagem de aprendizado *end-to-end* (Figura 13). De forma geral, todas as arquiteturas (DenseNet161, EfficientNetB3, ResNet50 e VGG19) apresentam elevadas acurácias de classificação por classe (i.e. poucas quantidades de amostras classificadas incorretamente, conforme observadas nas matrizes). As principais confusões ocorreram entre as classes C_8-C_7 (Strawberry - não padrão e Strawberry - padrão); C_6-C_5 (Candy - não padrão e Candy - padrão);

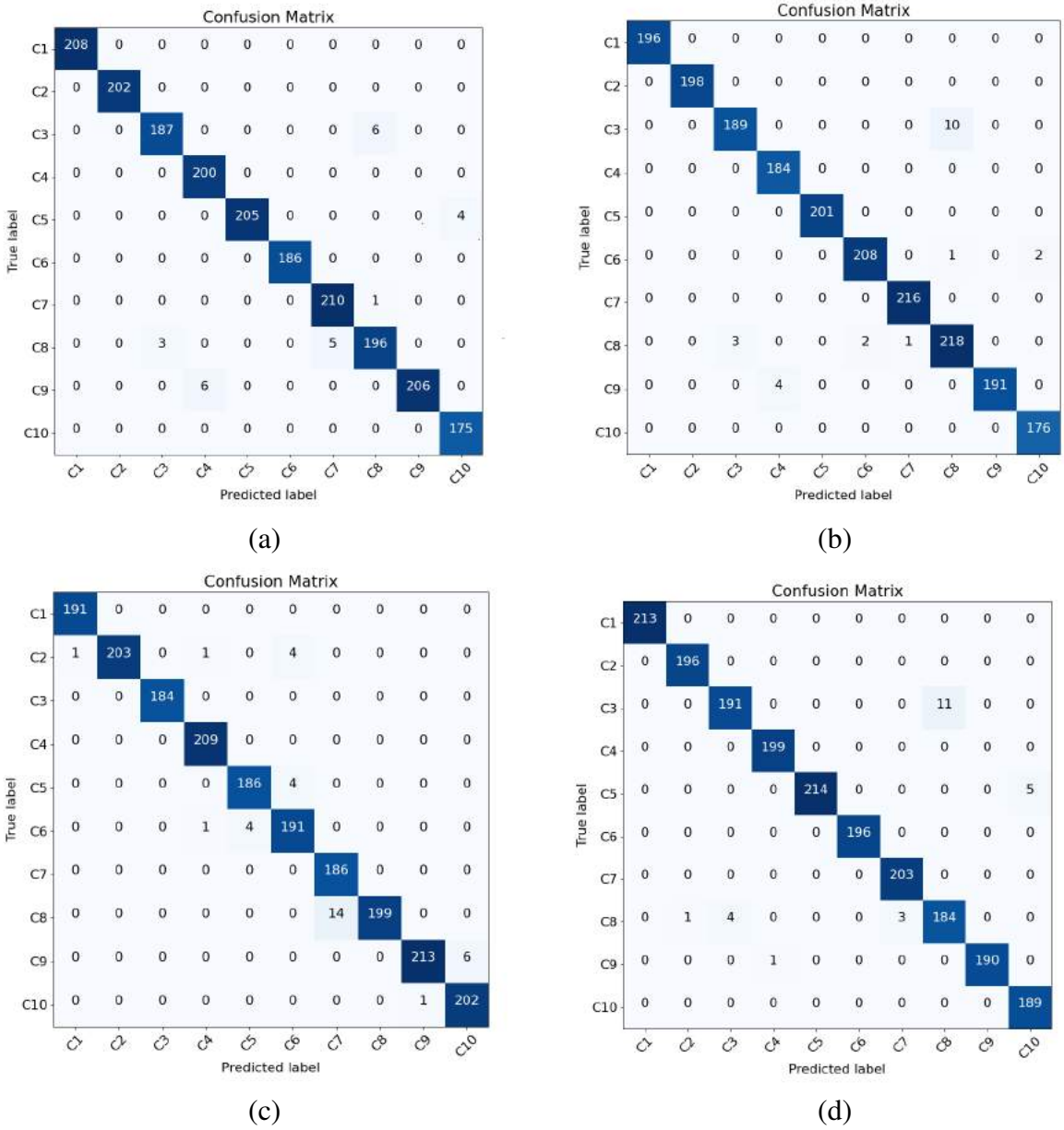


Figura 11: Matrizes de confusão obtidas pela abordagem de aprendizado tradicional utilizando as melhores combinações (pares descritor-classificador): (a) JCD-RF. (b) LCH-RF. (c) DenseNet161-*k*-NN. (d) DenseNet161-RF.

C_7-C_8 (Strawberry - padrão e Strawberry - não padrão). Sendo os piores desempenhos obtidos pela arquitetura ResNet50, entre as classes C_8-C_7 (19 amostras) e C_6-C_5 (10 amostras).

É possível observar que as principais confusões são encontradas em casos já esperados, i.e. entre classes de biscoitos que compartilham o mesmo sabor, diferenciando-se apenas por estarem ou não em conformidade ao padrão. Este comportamento é mais consistente ao ser comparado com as abordagens de aprendizado tradicional por meio dos pares descritores-classificadores, em que apresentam confusões não somente no padrão anteriormente mencionado, como também entre classes de sabores diferentes.

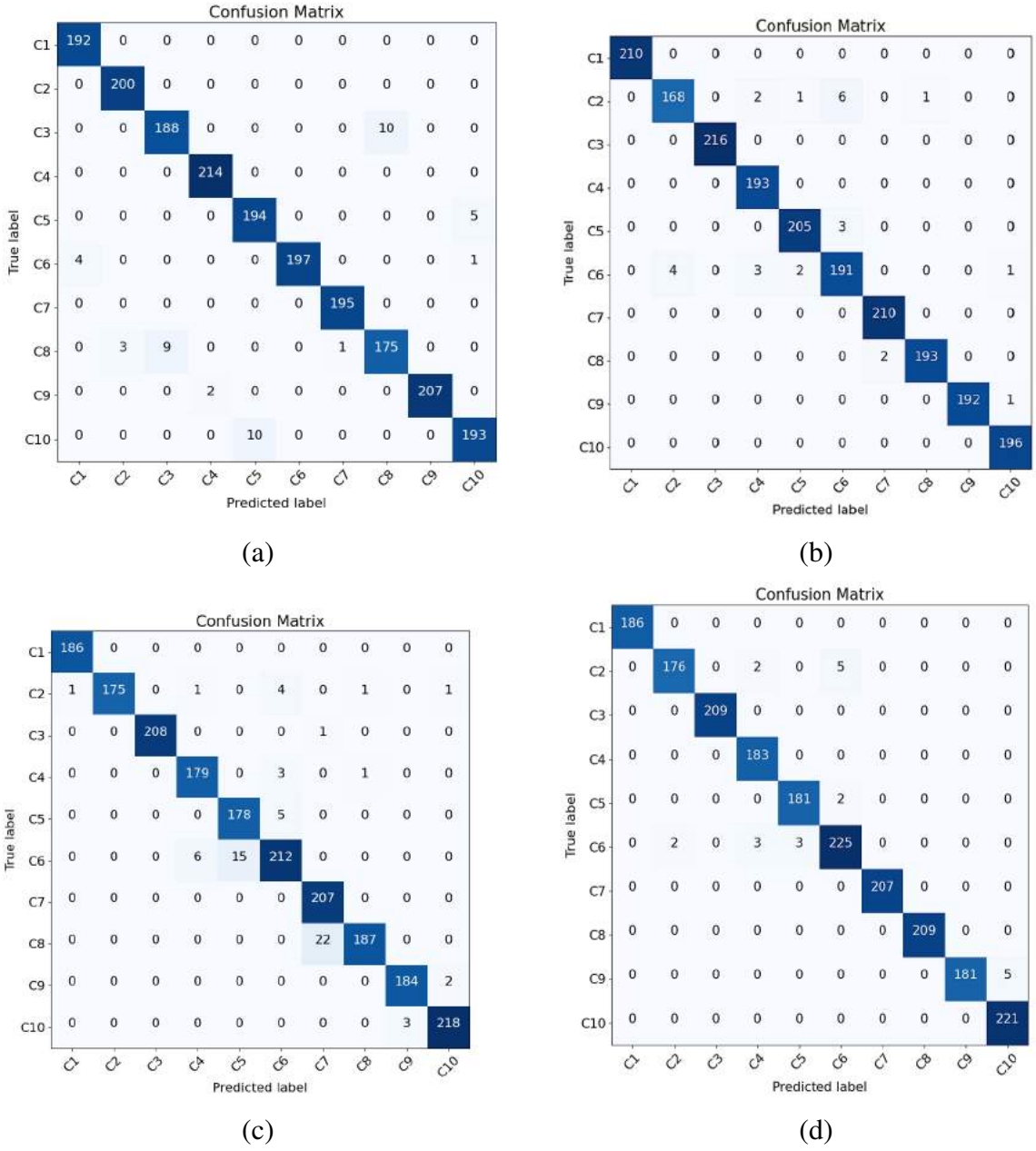


Figura 12: Matrizes de confusão obtidas pela abordagem de aprendizado tradicional utilizando as melhores combinações (pares descritor-classificador): (a) EfficientNetB3-*k*-NN. (b) EfficientNetB3-RF. (c) VGG19-*k*-NN. (d) VGG19-RF.

4.3.1 ANÁLISE GRAD-CAM

Além das análises realizadas considerando as métricas mencionadas anteriormente, é também apresentada a análise considerando o método *Gradient-weighted Class Activation Mapping* - Grad-CAM.

A Figura 14(a)-(j) apresenta exemplos de imagens resultantes da análise do Grad-CAM para cada uma das arquiteturas (DesnseNet161, EfficientNetB3, ResNet50, VGG19) e para cada

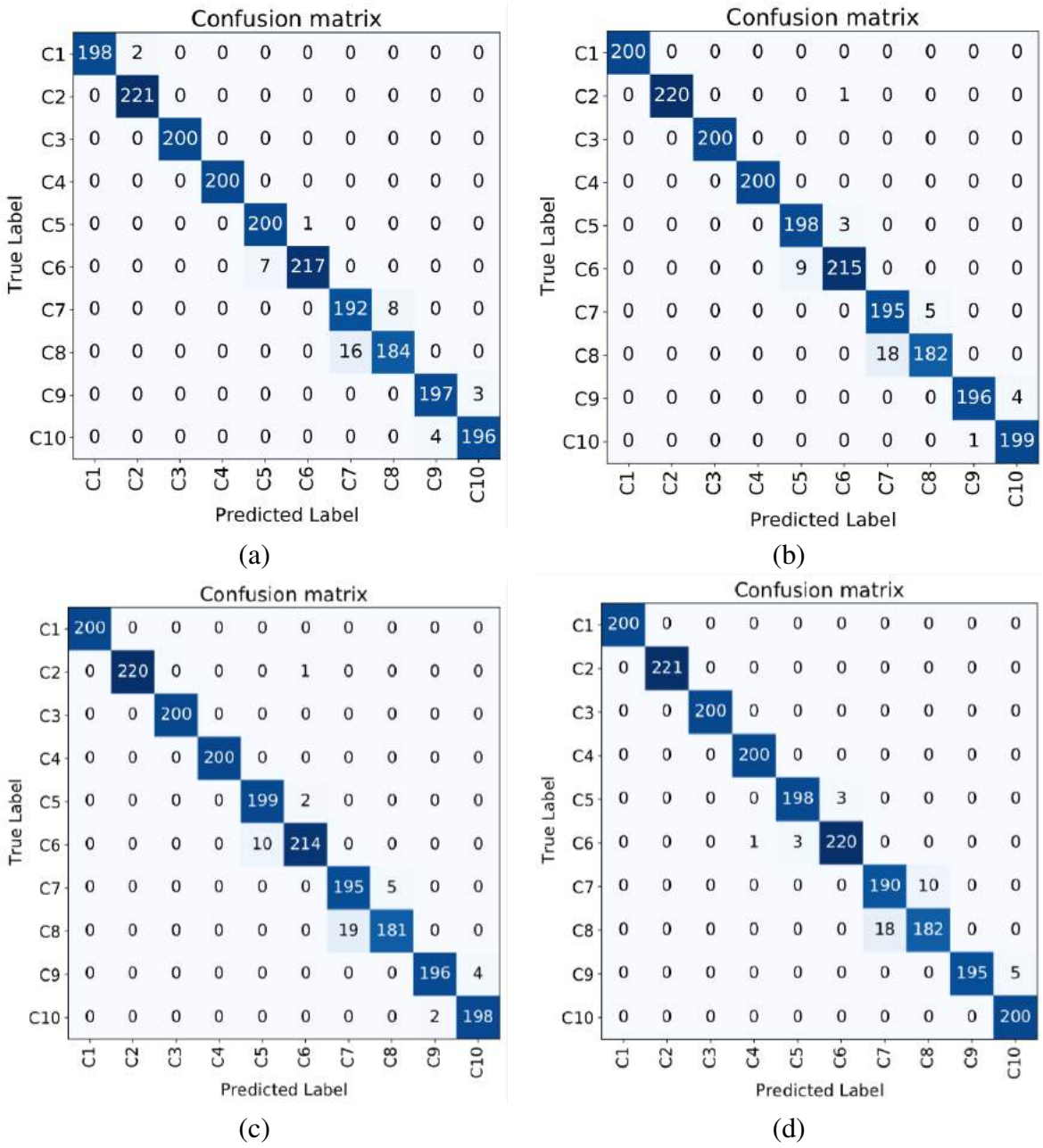


Figura 13: Matrizes de confusão obtidas pela abordagem *end-to-end* utilizando as arquiteturas CNNs: (a) DenseNet161. (b) EfficientNetB3. (c)ResNet50. (d) VGG19.

uma das classes C_1 a C_{10}, respectivamente. Nota-se que, considerando todas as arquiteturas, para a maioria das classes, os mapas são similares entre si, com exceção de algumas classes (Figuras 14(g), (h) e (i)), em que é possível observar facilmente diferenças tanto na forma quanto na localização das ativações.

Para a análise dos resultados considerando o Grad-CAM, é importante analisar onde as ativações da arquiteturas estão concentradas. Dessa forma, são apresentados exemplos de imagens menos confundidas (Figura 15) e de imagens mais confundidas (Figura 16) pela arquitetura DenseNet161, de modo a observar o que pode estar influenciando na decisão de classificação

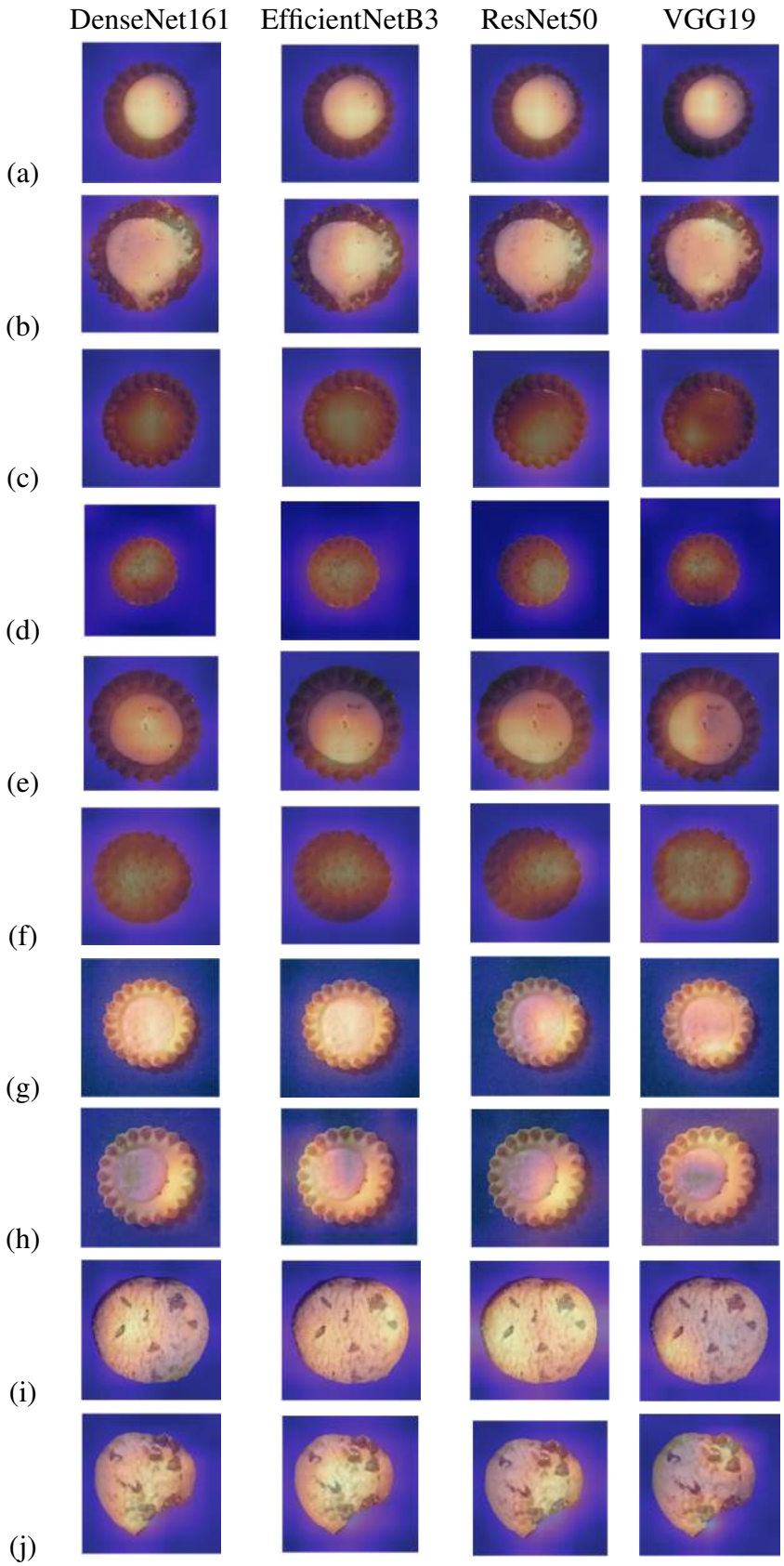


Figura 14: Exemplos de imagens resultantes da análise do Grad-CAM para as arquiteturas DenseNet161; EfficientNetB3; ResNet50 e VGG19, para cada uma das classes C_1 a C_{10} (a-j), respectivamente.

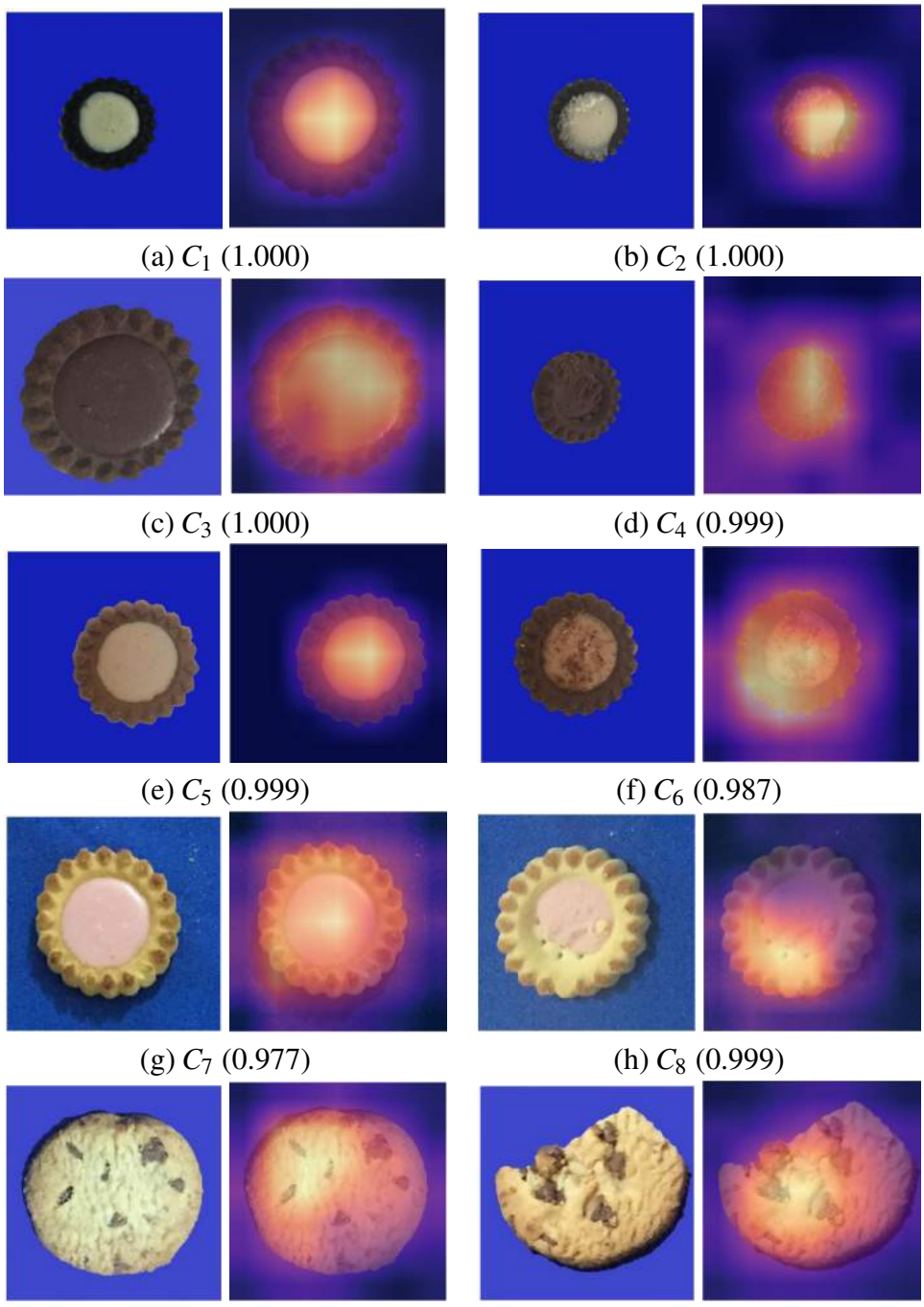
da arquitetura.

Na Figura 15(a)-(j) são apresentados exemplos de imagens, original (à esquerda) e resultante da análise do Grad-CAM (à direita), menos confundidas obtidas pela arquitetura DenseNet161, para cada uma das classes C_1 a C_{10}, respectivamente. São apresentadas as classes preditas e suas respectivas probabilidades entre parênteses. Verifica-se os mapas de ativação para as imagens que a arquitetura apresentou um alto grau de confiança (menor erro) em sua predição.Estas análises são úteis em relação à compreensão de funcionamento das CNNs, dado que a técnica Grad-CAM possibilita destacar áreas de interesse de uma determinada imagem que podem influenciar na decisão do modelo, o que auxilia também na depuração do processo de treinamento dessas arquiteturas.

Por exemplo, pode-se observar na Figura 15(h) que as ativações da arquitetura encontram-se principalmente concentradas em uma área em que não há recheio algum de biscoito. Sendo assim, a arquitetura conseguiu obter $99,9\%$ de probabilidade da imagem pertencer à classe C_8 (Strawberry - não padrão). O mesmo pode ser observado em diversos outros exemplos (e.g. nas Figuras 15(b), 15(d), 15(f), 15(j)), nos quais foi possível identificar as áreas específicas na imagem que representam a não conformidade aos padrões de qualidade dos biscoitos.

Já a Figura 16 ilustra exemplos de imagens, originais (à esquerda) e resultante da análise do Grad-CAM (ao centro e à direita), mais confundidas pela arquitetura DenseNet161. As imagens ao centro e à direita representam os mapas de ativação referentes às classes verdadeiras e preditas, respectivamente. São indicadas as classes confundidas (no formato classe verdadeira seguida da classe predita), bem como suas respectivas probabilidades entre parênteses.

Na Figura 16(e) tem-se a imagem original (à esquerda) da classe C_8 (Strawberry - não padrão) e as imagens (ao centro e à direita) relacionadas aos mapas de ativação referentes às classes verdadeira (C_8) e predita (C_7 – Strawberry - padrão), respectivamente. Nota-se que a imagem foi classificada como sendo da classe C_7 dada à probabilidade de $82,8\%$ (enquanto que a probabilidade para a classe C_8 foi de $17,2\%$. Pode-se observar que tal probabilidade foi obtida e, consequentemente, a classificação incorreta ocorreu devido às regiões ativadas na imagem. Nesse caso, a região ativada foi a parte superior esquerda da imagem, enquanto que a região da imagem que encontra-se de fato as características relacionadas a não conformidade aos padrões do biscoito encontram-se na parte superior direita da imagem (i.e. recheio ultrapassando a borda do biscoito, conforme pode ser visto na imagem original).



(a) C_1 (1.000) (b) C_2 (1.000)

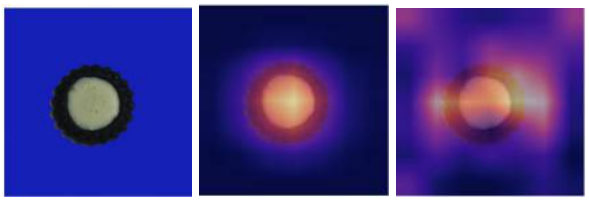
(c) C_3 (1.000) (d) C_4 (0.999)

(e) C_5 (0.999) (f) C_6 (0.987)

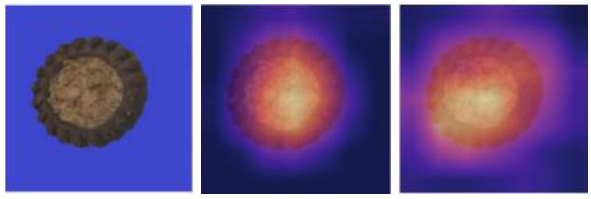
(g) C_7 (0.977) (h) C_8 (0.999)

(i) C_9 (1.000) (j) C_{10} (1.000)

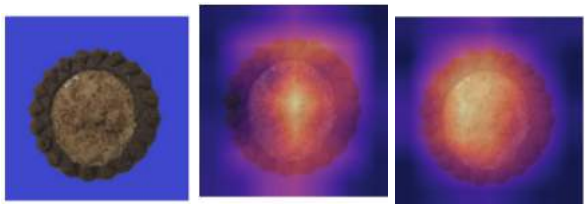
Figura 15: Exemplos de imagens, original (à esquerda) e resultante da análise do Grad-CAM (à direita), menos confundidas obtidas pela arquitetura DenseNet161, para cada uma das classes C_1 a C_{10} (a)-(j), respectivamente. São apresentadas as classes preditas e suas respectivas probabilidades entre parênteses.



(a) C_1-C_2 (0.154)-(0.846)



(b) C_5-C_6 (0.100)-(0.899)



(c) C_6-C_5 (0.481)-(0.519)

(d) C_7-C_8 (0.472)-(0.527)

(e) C_8-C_7 (0.172)-(0.828)

(f) C_9-C_{10} (0.000)-(1.000)

(g) C_{10}-C_9 (0.017)-(0.983)

Figura 16: Exemplos de imagens, originais (à esquerda) e resultantes da análise do Grad-CAM (ao centro e à direita), mais confundidas obtidas pela arquitetura DenseNet161. As imagens ao centro e à direita representam os mapas de ativação referentes às classes verdadeiras e preditas, respectivamente. São indicadas as classes confundidas (no formato classe verdadeira seguida da classe predita), bem como suas respectivas probabilidades entre parênteses.

5 CONSIDERAÇÕES FINAIS

Este projeto propõe uma metodologia envolvendo abordagens de aprendizado de descritores e de classificadores (SILVA et al., 2020), para aplicação em uma indústria do ramo alimentício, mais especificamente relacionada à análise e ao controle de qualidade na produção de biscoitos.

Para validar a metodologia proposta, foi realizada uma análise experimental extensiva, considerando diferentes abordagens de aprendizado. Para a abordagem de aprendizado tradicional foram realizados experimentos considerando diferentes tipos de características, obtidas por meio de extratores tradicionais e por meio das redes neurais profundas, aplicadas aos classificadores tradicionais. Para a abordagem de aprendizado *end-to-end* foram realizados experimentos considerando diferentes arquiteturas de redes neurais convolucionais.

Diferentes cenários foram considerados, envolvendo não apenas a classificação binária, i.e. identificação de biscoitos correspondentes (ou não) aos padrões exigidos pela empresa, mas também a classificação multiclasse, i.e. identificação de diferentes tipos de biscoitos. Isto se faz necessário, devido ao portfólio variado de produtos fabricados pela empresa, podendo inclusive apresentar padrões diferentes para o mesmo produto, dependendo da região do país onde o produto é comercializado. Um exemplo disto é o biscoito Cracker que na região Sudeste e Sul do país, o mesmo deve ser menos tostado, apresentando uma cor mais clara e, em contrapartida na região Nordeste, este mesmo produto deve ser mais tostado com uma cor mais escura. Este sendo um dos motivos de se encontrar uma técnica de classificação que possa abranger diferentes tipos de biscoitos e diferentes tipos de falhas.

Além disso, diferentes métricas de avaliação foram exploradas, tais como: acurácias geral e por classe, precisão, revocação, F1-Score, tempos computacionais, dimensionalidade dos vetores de características obtidos pelos descritores, visualização por meio da técnica Grad-CAM).

A partir dos resultados, pode-se observar que a metodologia proposta atinge acurácias de até 99%. Ambas as abordagens de aprendizado (tradicional e *end-to-end*) apresentam resul-

tados equivalentes. No entanto, apesar de apresentar determinados custos, devido à necessidade de aprendizados dos pesos para as arquiteturas das redes neurais convolucionais, a abordagem *end-to-end* não requer o estudo dos melhores descritores e classificadores, conforme a abordagem de aprendizado tradicional.

E em comparação ao processo aplicado atualmente na fábrica, em que as funcionárias mantêm-se posicionadas ao lado da esteira, separando os biscoitos não padrão, houve um ganho significativo. A classificação realizada pelas mesmas não chega a 15 amostras por minuto, ao contrário do processo de classificação apresentado neste trabalho em que foi possível classificar 220 amostras em 22 segundos durante os testes.

Como trabalhos futuros, seria interessante realizar outras análises relacionadas a outros tipos de biscoitos, bem como outros tipos de problemas encontrados nos biscoitos, além dos já considerados. Como por exemplo, mensurar a distribuição e a quantidade de gotas de chocolate no tipo de biscoito *cookies*; questões relacionadas a condições ambientais, como umidade e embalagem (aberta ou fechadas de forma adequada); entre outros. Para tanto, é necessária a obtenção de novas amostras de tipos diferentes de biscoitos e de problemas encontrados nos mesmos, bem como de uma quantidade maior de amostras de cada uma das classes consideradas no presente projeto, de forma a melhorar o processo de aprendizado. No entanto, existem algumas dificuldades relacionadas ao processo de aquisição das imagens, dada à necessidade de um funcionário à disposição para obtenção das imagens.

Trabalhos futuros também envolvem explorar e investigar outras técnicas mais eficazes e eficientes para melhorar a descrição e a classificação, incluindo, por exemplo, a aplicação de estratégias de aprendizado ativo e semi-supervisionado, de forma a acelerar o processo de aprendizado dos classificadores.

REFERÊNCIAS

ABADI, M. et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. **CoRR**, abs/1603.04467, 2016. Disponível em: <http://arxiv.org/abs/1603.04467>.

ALVAREZ, D. et al. Usefulness of artificial neural networks in the diagnosis and treatment of sleep apnea-hypopnea syndrome. IntechOpen, Rijeka, p. 33–68, 04 2017. Disponível em: <https://doi.org/10.5772/66570>.

ARWAN, A. Determining basis test paths using genetic algorithm and j48. **International Journal of Electrical and Computer Engineering**, v. 8, n. 5, p. 3333–3340, 2018.

BISHOP, C. M. **Pattern Recognition and Machine Learning (Information Science and Statistics)**. 1. ed. New York: Springer, 2006. 738 p. ISBN 0387310738.

BREIMAN, L. Random forests. **Machine Learning**, Kluwer Academic Publishers, v. 45, n. 1, p. 5–32, 2001. ISSN 0885-6125. Disponível em: <https://doi.org/10.1023/A:1010933404324>.

BRESSAN, R. S.; SAITO, P. T. M. Aprendizado ativo para recuperação e classificação de imagens. **CP - Programa de Pós-Graduação em Informática**, Universidade Tecnológica Federal do Paraná, p. 86, 2018. Disponível em: <http://repositorio.utfpr.edu.br/jspui/handle/1/4534>.

BRILHADOR, A. **Combinando Descritores de Forma e Textura para classificação de Bioimagens:Um Estudo de caso aplicado a base de imagens imageclef**. 2014. Disponível em: <http://paginapessoal.utfpr.edu.br/fabricio/pesquisa/CP_PPGI_M_Brilha dorAnderson_2015.pdf/view>.

BROSNAN, T.; SUN, D.-W. Improving quality inspection of food products by computer vision—a review. **Journal of Food Engineering**, v. 61, p. 3–16, 01 2004.

CHATZICHRISTOFIS, S. A.; BOUTALIS, Y. S. Cedd: Color and edge directivity descriptor: A compact descriptor for image indexing and retrieval. In: GASTERATOS, A.; VINCZE, M.; TSOTSOS, J. K. (Ed.). **ICVS**. Springer, 2008. (Lecture Notes in Computer Science, v. 5008), p. 312–322. ISBN 978-3-540-79546-9. Disponível em: <http://dblp.uni-trier.de/db/conf/icvs/icvs2008.html#ChatzichristofisB08>.

CHATZICHRISTOFIS, S. A.; BOUTALIS, Y. S. Fcth: Fuzzy color and texture histogram - a low level feature for accurate image retrieval. In: **WIAMIS**. IEEE Computer Society, 2008. p. 191–196. ISBN 978-0-7695-3130-4. Disponível em: <http://dblp.uni-trier.de/db/conf/wiamis/wiamis2008.html#ChatzichristofisB08>.

CUBEDDU, A.; RAUH, C.; DELGADO, A. Hybrid artificial neural network for prediction and control of process variables in food extrusion. **Innovative Food Science & Emerging Technologies**, v. 21, p. 142 – 150, 2014. ISSN 1466-8564. Disponível em: <http://www.sciencedirect.com/science/article/pii/S146685641300163X>.

DAVIDSON, V. J.; RYKS, J.; CHU, T. Fuzzy models to predict consumer ratings for biscuits based on digital image features. **IEEE Transactions on Fuzzy Systems**, v. 9, n. 1, p. 62–67, 2001. ISSN 1941-0034.

GONZALEZ, R. C.; WOODS, R. E. **Digital Image Processing**. 2nd. ed. USA: Addison-Wesley Longman Publishing Co., Inc., 2001. ISBN 0201180758.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. MIT Press, 2016. 436-44 p. (Adaptive computation and machine learning, v. 521). ISBN 9780262035613. Disponível em: <https://books.google.co.in/books?id=Np9SDQAAQBAJ>.

GUNAYDIN, H. Probabilistic approach to generating mpos and its application as a scoring function for cns drugs. **ACS Medicinal Chem. Lett.**, v. 7, n. 1, p. 89–93, 2016. Disponível em: <https://doi.org/10.1021/acsmedchemlett.5b00390>.

HE, Z. et al. Semisupervised svm based on cuckoo search algorithm and its application. **Mathematical Problems in Engineering**, v. 2018, p. 1–13, 2018. Disponível em: <https://doi.org/10.1155/2018/8243764>.

HELMAN, H. Anuario abimapi 2019. **Anuario Abimapi**, 2019. Disponível em: <https://www.abimapi.com.br/anuario.php>.

HOWARD, J.; GUGGER, S. Fastai: A layered api for deep learning. **Information**, MDPI AG, v. 11, n. 2, p. 108, Feb 2020. ISSN 2078-2489. Disponível em: <http://dx.doi.org/10.3390/info11020108>.

HUANG, D. et al. Local binary patterns and its application to facial image analysis: A survey. **IEEE Trans. Syst. Man Cybern. Part C**, v. 41, n. 6, p. 765–781, 2011. Disponível em: <http://dblp.uni-trier.de/db/journals/tsmc/tsmcc41.html#HuangSAWC11>.

HUANG, G. et al. **Densely Connected Convolutional Networks**. 2016. 2261-2269 p. Cite arxiv:1608.06993Comment: CVPR 2017. Disponível em: <http://arxiv.org/abs/1608.06993>.

JACKMAN, P.; SUN, D.-W. Recent advances in image processing using image texture features for food quality assessment. **Trends in Food Science & Technology**, v. 29, p. 35–43, 01 2013. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0924224412001896>.

JURAN, J. M. **Juran's Quality Control Handbook**. 4th. ed. Mcgraw-Hill, 1974. 1774 p. Hardcover. ISBN 0070331766. Disponível em: <http://www.amazon.com>.

KARMAKAR, P. et al. Improved tamura features for image classification using kernel based descriptors. **Intl. Conf. on Digital Image Computing: Techniques and Applications**, p. 1–7, 2017. Disponível em: <https://ieeexplore.ieee.org/document/8227447>. Acesso em: 29 de Novembro de 2017.

KINGMA, D. P.; BA, J. **Adam: A Method for Stochastic Optimization**. 2014. Cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015. Disponível em: <http://arxiv.org/abs/1412.6980>.

KUMAR, P.; D.APARNA; RAO, K. V. Compact descriptors for accurate image indexing and retrieval: Fcth and cedd. **International journal of engineering research and technology**, v. 1, p. 11, 2012. ISSN 2278-0181. Disponível em: <https://www.ijert.org/>.

LEMANZYK, T. et al. Food safety by using machine learning for automatic classification of seeds of the south-american incanut plant. **Journal of Physics: Conference Series**, IOP Publishing, v. 588, p. 012036, 2015. Disponível em: <https://doi.org/10.1088/1742-6596/588/1/012036>.

LU, Y. Food image recognition by using convolutional neural networks (cnns). **CoRR**, abs/1612.00983, 2016. Disponível em: <http://arxiv.org/abs/1612.00983>.

LöFSTEDT, T. et al. Gray-level invariant haralick texture features. **PLOS ONE**, Public Library of Science, v. 14, n. 2, p. 1–18, 2019. Disponível em: <https://doi.org/10.1371/journal.pone.0212110>.

MAHMOOD, A. et al. Automatic hierarchical classification of kelps using deep residual features. **Sensors**, v. 20, n. 2, p. 447, 2020. Disponível em: <http://dblp.uni-trier.de/db/journals/sensors/sensors20.html#MahmoodOBASBHFK20>.

MARTINS, P. G.; LAUGENI, F. P. **Administração da Produção**. New York: Editora Saraiva, 2005. 584 p. ISBN 9788502618350.

MITRA, M.; ZHU, W.-j.; ZABIH, R. **Image Indexing Using Color Correlograms**. 2002. 762-768 p. Disponível em: <https://ieeexplore.ieee.org/document/609412>. Acesso em: 06 de Agosto de 2018.

MOSAVI, A. Deep learning: a review. **Advances in Intelligent Systems and Computing**, v. 1, p. 11, 2017. Disponível em: <https://www.researchgate.net/publication/328285467>.

NASHAT, S.; ABDULLAH, A.; ABDULLAH, M. Machine vision for crack inspection of biscuits featuring pyramid detection scheme. **Journal of Food Engineering**, v. 120, p. 233–247, 2014. ISSN 0260-8774. Disponível em: <http://www.sciencedirect.com/science/article/pii/S026087741300410X>.

NASHAT, S. et al. Support vector machine approach to real-time inspection of biscuits on moving conveyor belt. **Computers and Electronics in Agriculture**, Elsevier, v. 75, n. 1, p. 147–155, 1 2011. ISSN 0168-1699.

RANI, P.; VASHISHTHA, J. An appraise of knn to the perfection. **International Journal of Computer Applications**, Foundation of Computer Science, v. 170, n. 2, p. 13–17, 2017. ISSN 0975-8887. Disponível em: <http://www.ijcaonline.org/archives/volume170/number2/28041-2017914696>.

RUBMANN, M. et al. **Industry 4.0: The Future of Productivity and Growth in Manufacturing Industries**. Boston: The Boston Consulting Group, 2015. 239-242 p. ISBN 9783935089296. Disponível em: <https://arxiv.org/>.

SAMMUT, C.; WEBB, G. I. (Ed.). **Encyclopedia of Machine Learning**. Springer, 2010. ISBN 978-0-387-30768-8. Disponível em: <http://dx.doi.org/10.1007/978-0-387-30164-8>.

SELVARAJU, R. R. et al. Grad-cam: Visual explanations from deep networks via gradient-based localization. **International Journal of Computer Vision**, Springer Science and Business Media LLC, v. 128, n. 2, p. 336–359, Oct 2019. ISSN 1573-1405. Disponível em: <http://dx.doi.org/10.1007/s11263-019-01228-7>.

SELVARAJU, R. R. et al. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. **CoRR**, abs/1610.02391, 2016. Disponível em: <http://arxiv.org/abs/1610.02391>.

SILVA, M. S. et al. Automatic visual quality assessment of biscuits using machine learning. In: **ICAISC 2020 - International Conference on Artificial Intelligence and Soft Computing**. [s.n.], 2020. p. 1–12. Disponível em: <http://www.icaisc.eu/>.

SIMONYAN, K.; ZISSERMAN, A. **Very Deep Convolutional Networks for Large-Scale Image Recognition**. 2014. Cite arxiv:1409.1556. Disponível em: <http://arxiv.org/abs/1409.1556>.

SIVAKUMAR, B.; SRILATHA, K. A survey on computer vision technology for food quality evaluation. **Research Journal of Pharmaceutical, Biological and Chemical Sciences**, v. 7, p. 365–373, 2016. Disponível em: <https://www.researchgate.net/publication/312494475>.

SMITH, L. N. **A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay**. 2018. Cite arxiv:1803.09820. Disponível em: <http://arxiv.org/abs/1803.09820>.

SRIVASTAVA, S.; BOYAT, S.; SADISTAP, S. A robust machine vision algorithm development for quality parameters extraction of circular biscuits and cookies digital images. **Journal of Food Processing**, v. 2014, p. 13, 2014. ISSN 2356-7384.

SZEGEDY, C. et al. Rethinking the inception architecture for computer vision. **CoRR**, abs/1512.00567, 2015. Disponível em: <http://arxiv.org/abs/1512.00567>.

TAN, M.; LE, Q. EfficientNet: Rethinking model scaling for convolutional neural networks. In: CHAUDHURI, K.; SALAKHUTDINOV, R. (Ed.). Long Beach, California, USA: PMLR, 2019. (Proceedings of Machine Learning Research, v. 97), p. 6105–6114. Disponível em: <http://proceedings.mlr.press/v97/tan19a.html>.

VARGAS, A. C. G.; PAES, A.; VASCONCELOS, C. N. **Um Estudo sobre Redes Neurais Convolucionais e sua Aplicação em Detecção de Pedestres**. 2016. Disponível em: <http://sibgrapi.sid.inpe.br/rep/sid.inpe.br/sibgrapi/2016>.

VINAYAK, V.; JINDAL, S. Cbir system using color moment and color auto-correlogram with block truncation coding. **International Journal of Computer Applications**, Foundation of Computer Science, v. 161, n. 9, p. 1–7, 2017. ISSN 0975-8887. Disponível em: <http://www.ijcaonline.org/archives/volume161/number9/27173-2017913282>.

WANG, J. et al. Deep learning for smart manufacturing: Methods and applications. **Journal of Manufacturing Systems**, v. 48, p. 144 – 156, 2018. ISSN 0278-6125. Special Issue on Smart Manufacturing. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0278612518300037>.

WANG, J. et al. Evaluation of color similarity descriptors for human action recognition. In: **Intl. Conf. on Internet Multimedia Computing and Service**. ACM, 2014. p. 197–200. ISBN 9781450328104. Disponível em: <https://doi.org/10.1145/2632856.2632858>.

WANG, S. **A Robust CBIR Approach Using Local Color Histograms [microform]**. Thesis (M.Sc.)–University of Alberta, 2001. ISBN 9780612696112. Disponível em: <https://books.google.com.br/books?id=A507GwAACAAJ>.

ZHANG, H.; SHA, Z. **Product Classification based on SVM and PHOG Descriptor**. 2013. Disponível em: <https://pdfs.semanticscholar.org/b5d2>.

ZHOU, B. et al. Learning deep features for discriminative localization. **CoRR**, abs/1512.04150, 2015. Disponível em: <http://arxiv.org/abs/1512.04150>.

APÊNDICE A – RESULTADOS

Tabela 6: Resultados referentes às diferentes métricas (precisão, revocação, F1-Score, tempos de treinamento e de teste (em segundos)) obtidas na abordagem de aprendizado tradicional pelos melhores pares descritor-classificador (incluindo os resultados equivalentes obtidos).

Descritor-Classificador	Precisão	Revocação	F1-Score	Tempo Teste	Tempo Treinamento
ACC-J48	97,51±0,31	97,47±0,33	97,46±0,33	6,08±0,74	146,26±7,23
BIC-J48	97,65±0,30	97,72±0,29	97,68±0,30	3,45±0,20	27,18±0,36
CEDD-J48	97,11±0,09	97,16±0,09	97,12±0,09	3,55±0,34	8,44±0,13
DenseNet161-J48	97,58±0,23	97,48±0,26	97,52±0,26	10,61±0,46	14,22±0,31
DenseNet161-SVM	97,57±0,23	97,50±0,25	97,51±0,25	30,94±0,58	87,20±1,03
EfficienteNetB3-J48	**98,78±0,02**	**98,87±0,05**	**98,88±0,06**	22,93±0,41	290,21±5,45
EfficienteNetB3-SVM	97,66±0,85	**97,94±0,93**	98,24±0,86	22,83±0,41	32,97±0,31
FCTH-J48	96,26±0,25	96,20±0,30	96,06±0,29	3,58±0,11	5,37±0,39
FCTH-RF	96,9±0,32	96,82±0,29	96,84±0,35	30,48±0,55	92,19±4,24
Gabor-RF	93,97±0,23	94,24±0,23	94,05±0,23	27,53±0,39	338,97±0,63
GCH-J48	97,81±0,47	97,79±0,47	97,79±0,48	3,04±0,09	38,07±0,94
Haralick-J48	94,84±0,17	94,91±0,16	94,83±0,16	**2,72±0,07**	5,13±0,03
Haralick-RF	96,29±0,12	96,29±0,12	95,88±0,70	26,92±4,59	122,22±0,32
JCD-J48	97,66±0,26	97,70±0,25	97,65±0,26	3,37±0,17	9,76±0,09
LBP-RF	95,69±0,21	95,59±0,21	95,54±0,22	28,95±0,41	595,5±3,16
LCH-J48	95,95±0,29	96,13±0,28	96,00±0,29	3,79±0,17	133,04±1,58
Moments-J48	97,01±0,25	96,95±0,16	96,75±0,67	**2,75±0,12**	**2,53±0,72**
Moments-RF	97,5±0,08	97,51±0,08	97,49±0,08	23,24±1,91	92,73±0,72
MPO-RF	97,23±0,14	97,26±0,16	97,24±0,14	24,72±0,37	120,47±5,75
PHOG-RF	93,5±0,22	93,66±0,20	93,49±0,22	28,16±0,48	335,37±3,64
RCS-RF	98,23±0,12	98,21±0,11	98,21±0,11	25,40±0,47	387,91±4,71
ResNet50-RF	98,31±0,15	98,33±0,14	98,31±0,15	55,23±1,23	901,79±3,16
Tamura-RF	95,64±0,16	95,68±0,17	95,59±0,17	26,57±0,46	167,64±0,63

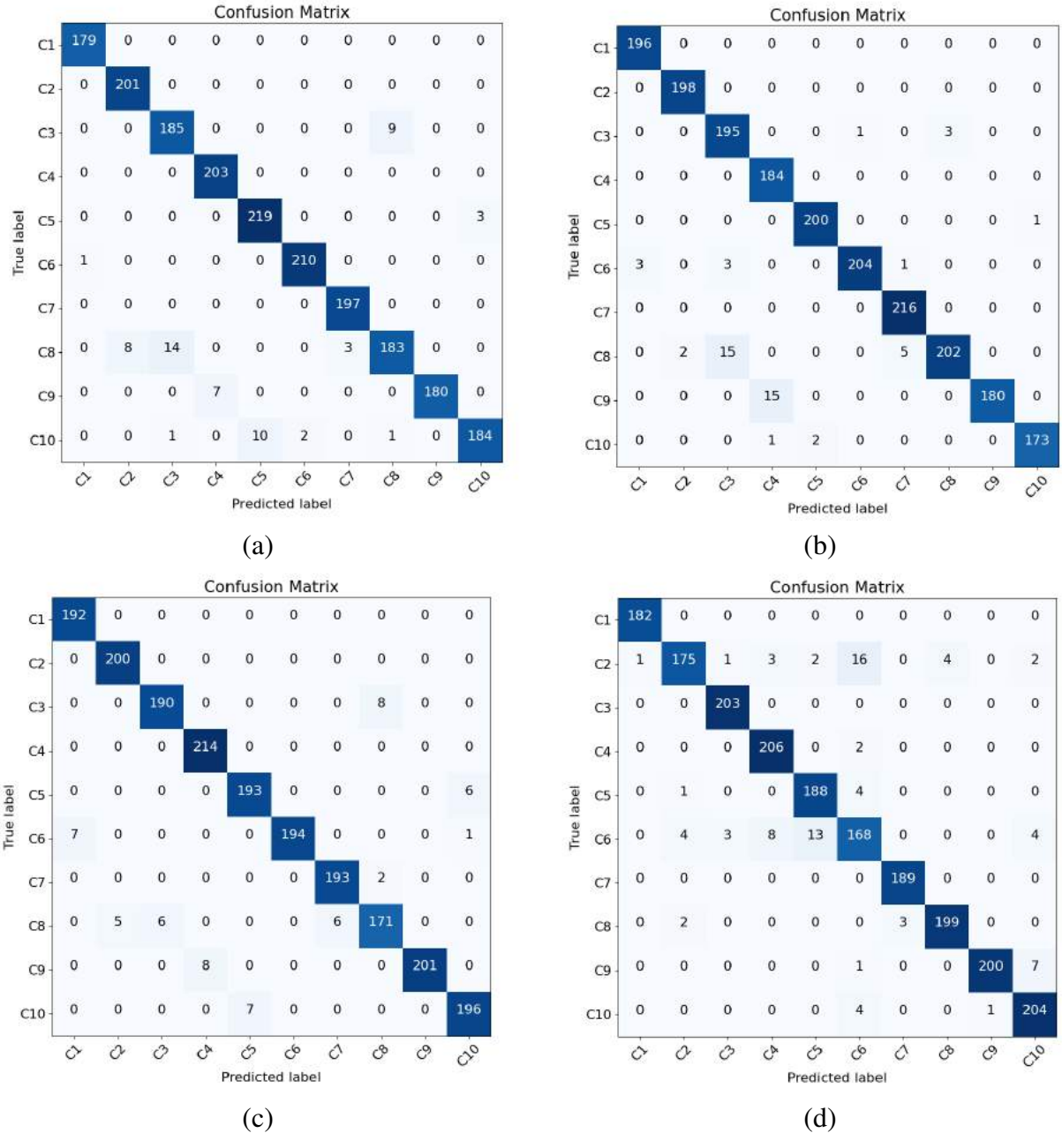


Figura 17: Matrizes de confusão obtidas pela abordagem de aprendizado tradicional utilizando as melhores combinações (pares descritor-classificador): (a) ACC-J48. (b) BIC-J48. (c) CEDD-J48. (d) DenseNet161-J48.

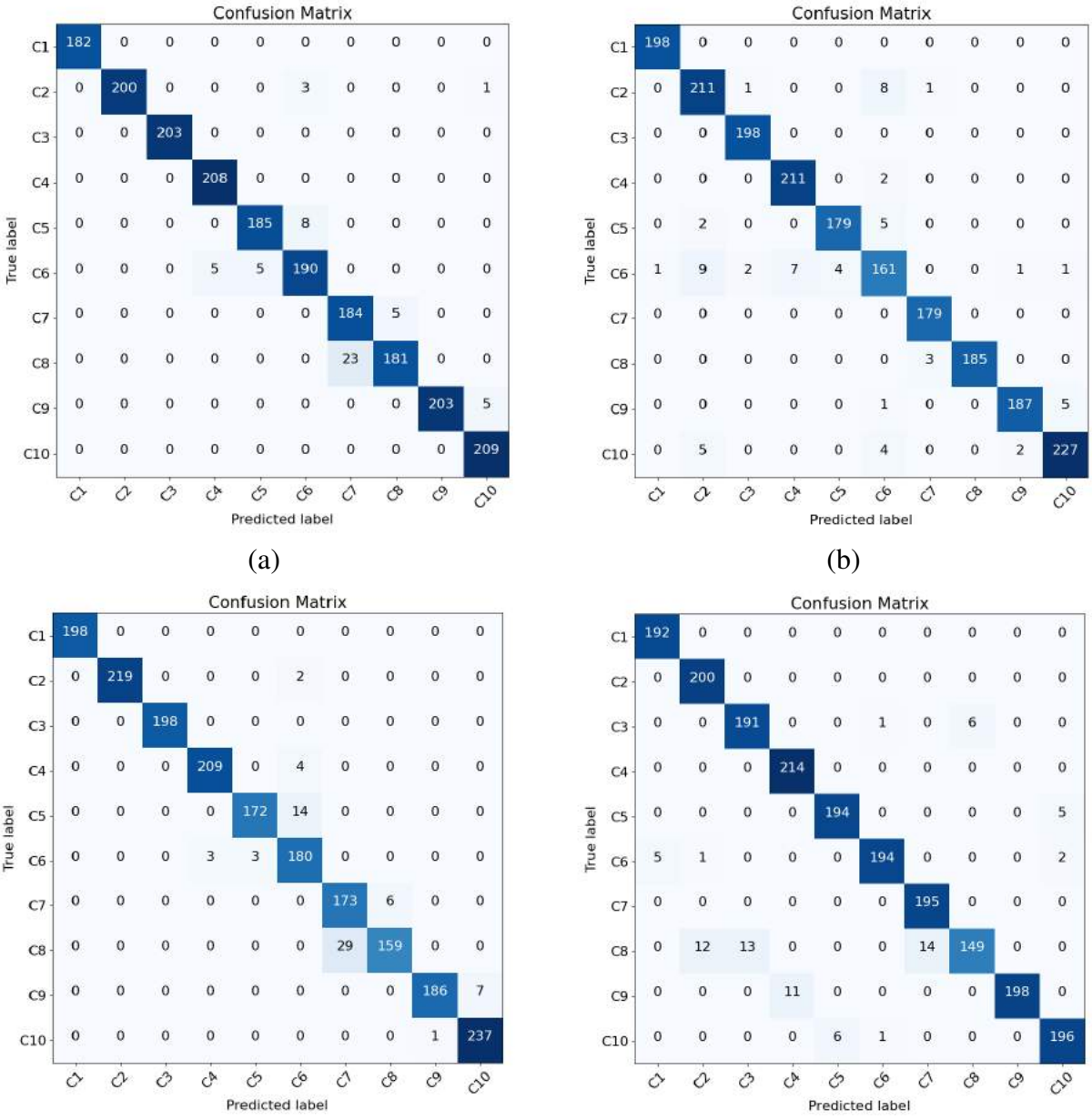


Figura 18: Matrizes de confusão obtidas pela abordagem de aprendizado tradicional utilizando as melhores combinações (pares descritor-classificador): (a) DenseNet161-SVM. (b) EfficienteNetB3-J48. (c) EfficienteNetB3-SVM. (d) FCTH-J48.

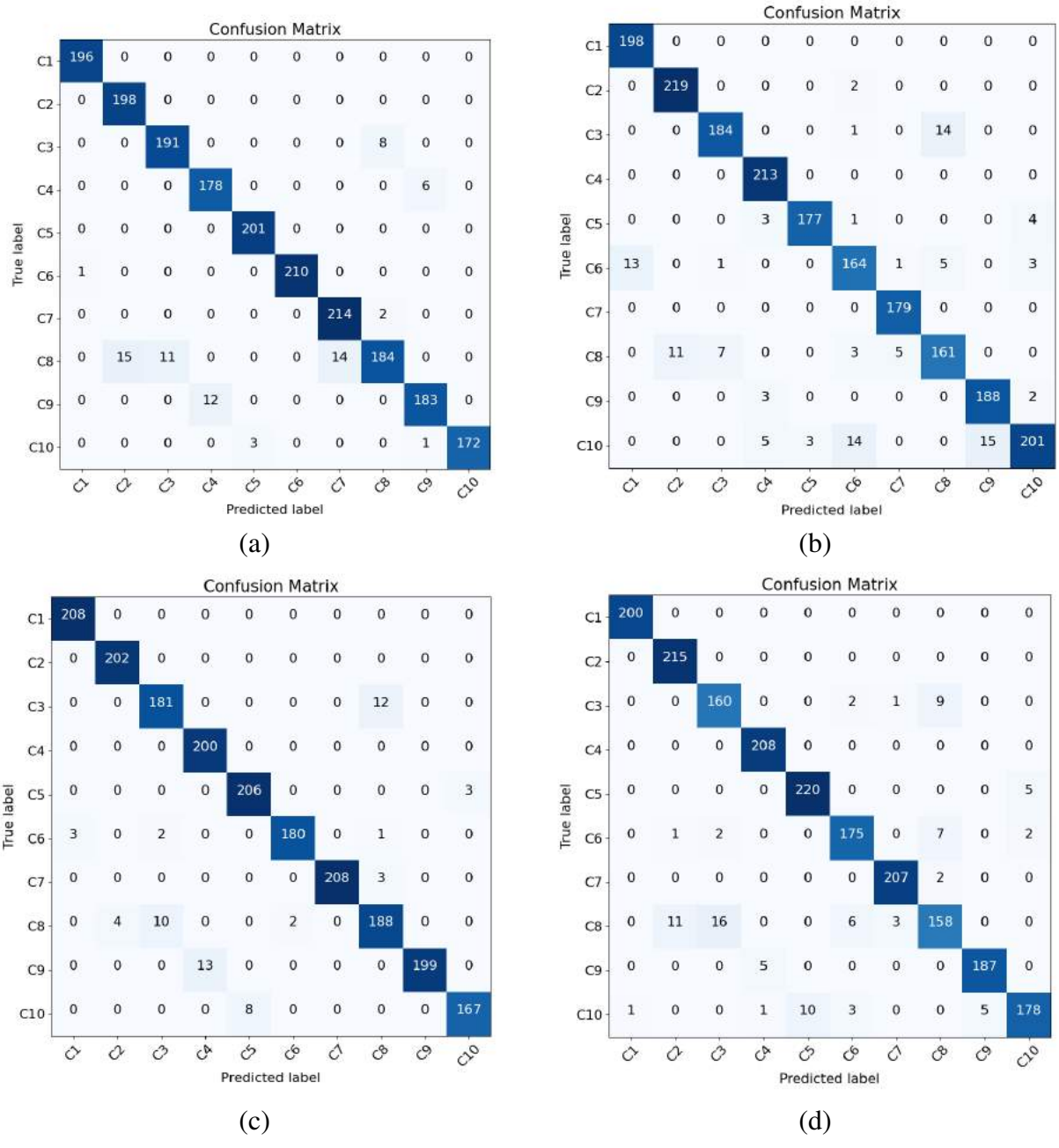


Figura 19: Matrizes de confusão obtidas pela abordagem de aprendizado tradicional utilizando as melhores combinações (pares descritor-classificador): (a) FCTH-RF. (b) Gabor-RF. (c) GCH-J48. (d) Haralick-J48.

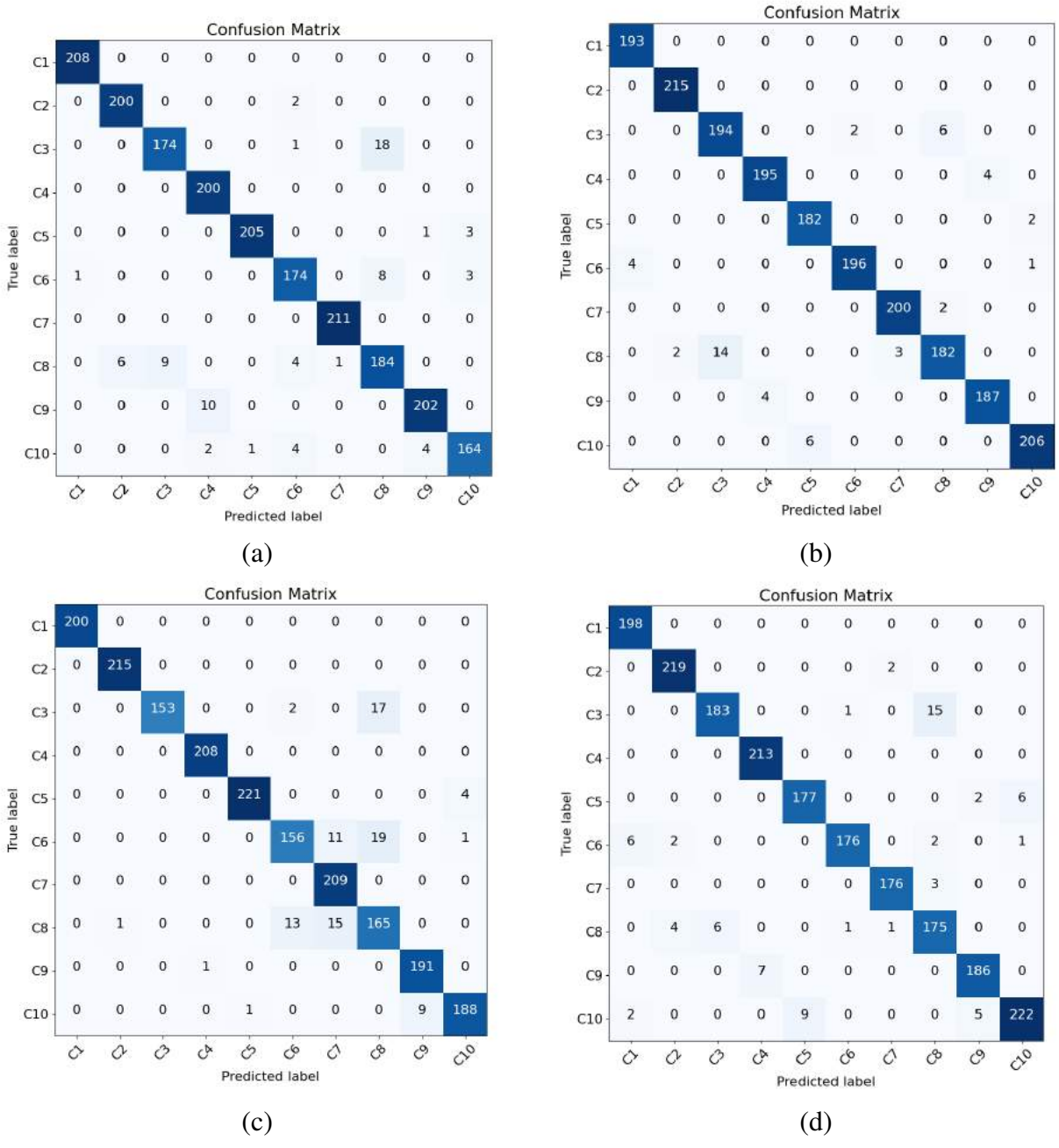


Figura 20: Matrizes de confusão obtidas pela abordagem de aprendizado tradicional utilizando as melhores combinações (pares descritor-classificador): (a) Haralick-RF. (b) JCD-J48. (c) LBP-RF. (d) LCH-J48.

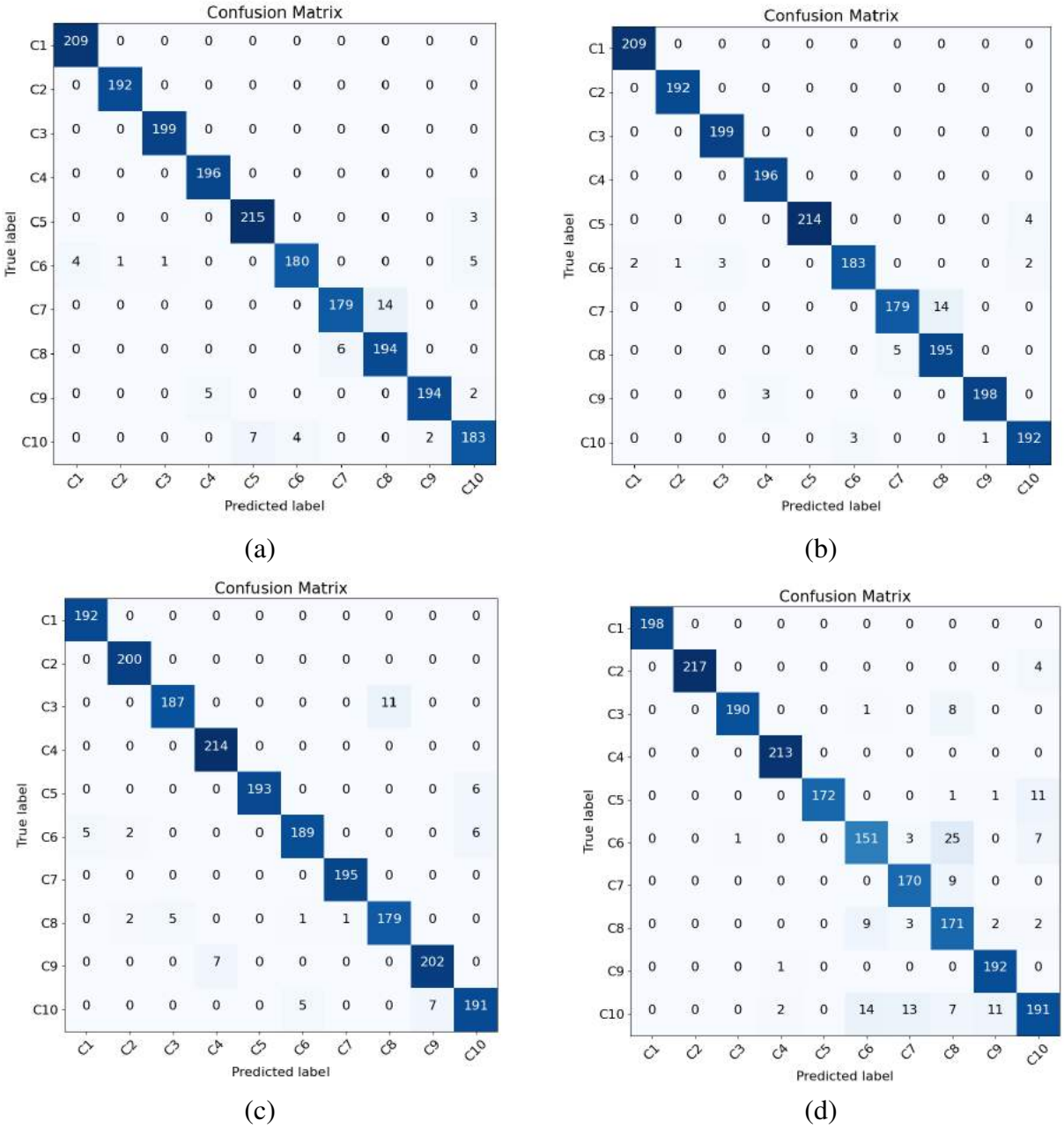


Figura 21: Matrizes de confusão obtidas pela abordagem de aprendizado tradicional utilizando as melhores combinações (pares descritor-classificador): (a) Moments-J48. (b) Moments-RF. (c) MPO-RF. (d) PHOG-RF.

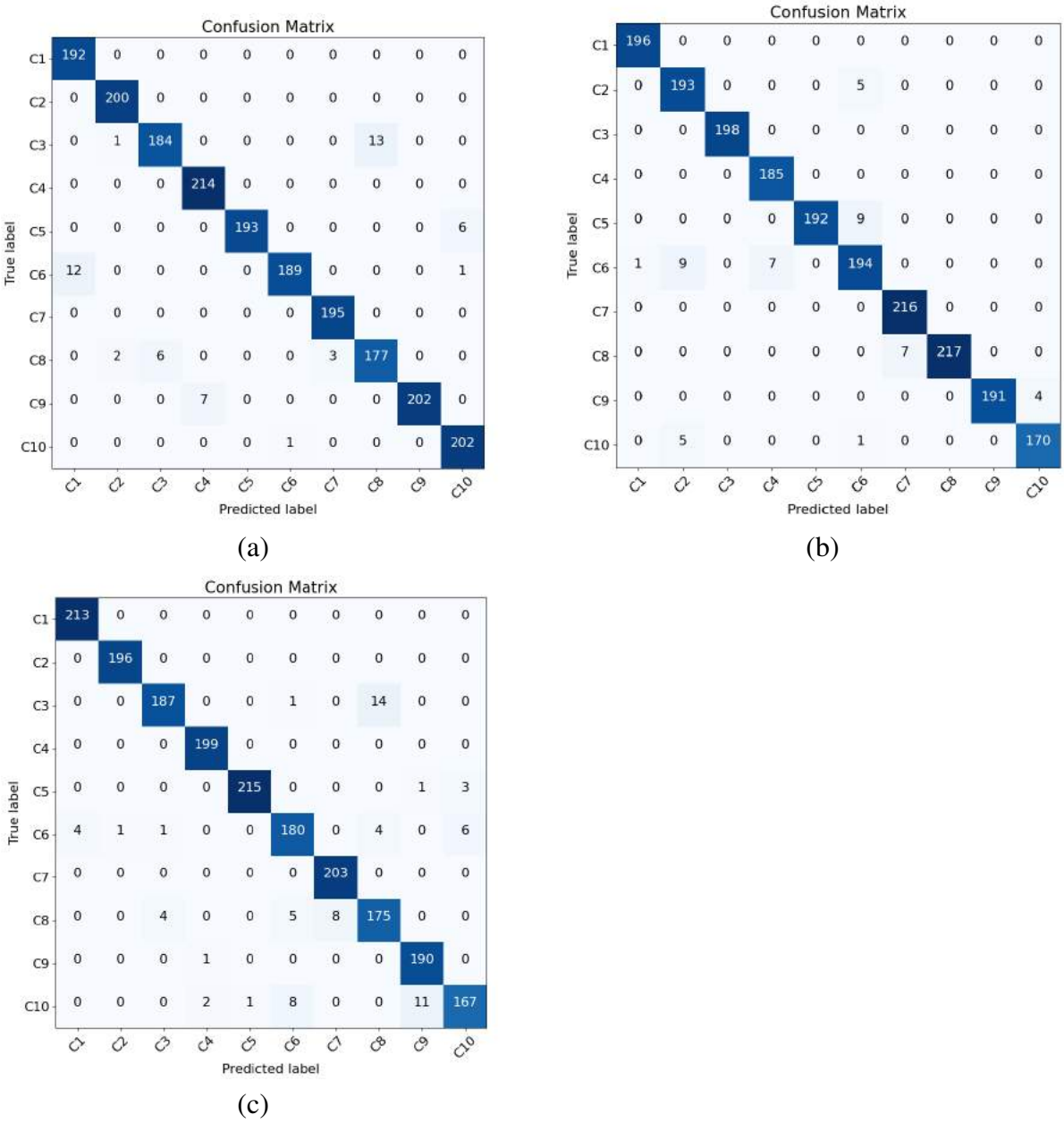
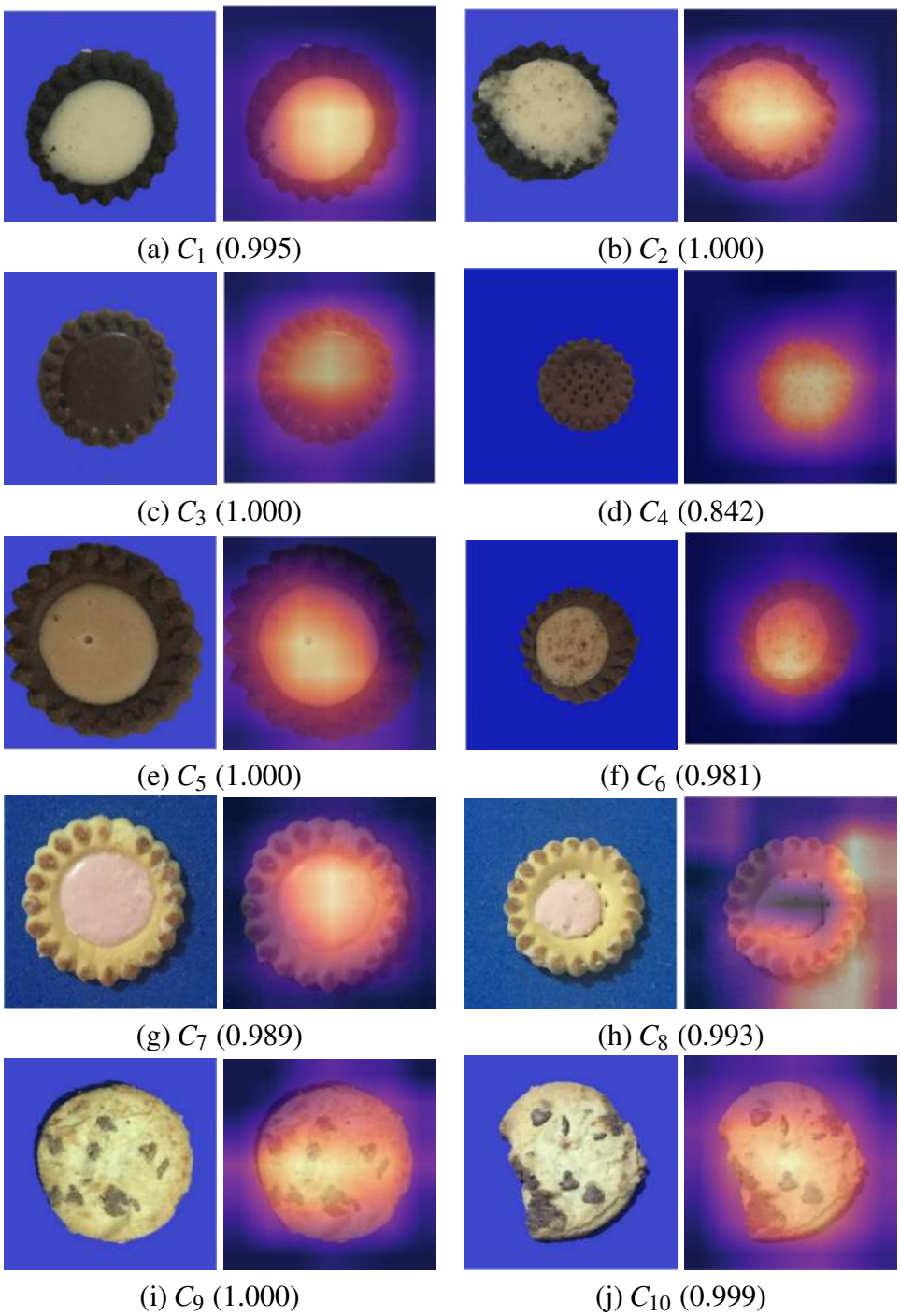


Figura 22: Matrizes de confusão obtidas pela abordagem de aprendizado tradicional utilizando as melhores combinações (pares descritor-classificador): (a) RCS-RF. (b) ResNet50-RF. (c) Tamura-RF.



(a) C_1 (0.995) (b) C_2 (1.000)

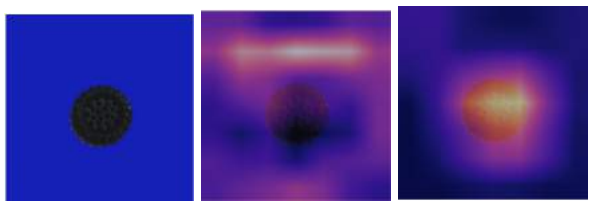
(c) C_3 (1.000) (d) C_4 (0.842)

(e) C_5 (1.000) (f) C_6 (0.981)

(g) C_7 (0.989) (h) C_8 (0.993)

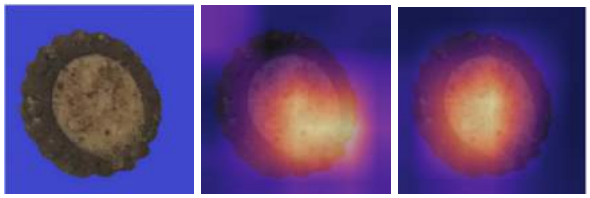
(i) C_9 (1.000) (j) C_{10} (0.999)

Figura 23: Exemplos de imagens, original (à esquerda) e resultante da análise do Grad-CAM (à direita), menos confundidas obtidas pela arquitetura EfficientNetB3, para cada uma das classes C_1 a C_{10} (a)-(j), respectivamente. São apresentadas as classes preditas e suas respectivas probabilidades entre parênteses.



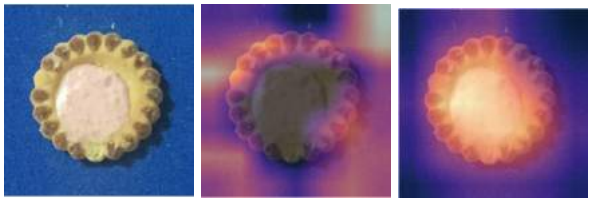
(a) C_2-C_6 (0.006)-(0.901)

(b) C_5-C_6 (0.076)-(0.924)



(c) C_6-C_5 (0.251)-(0.749)

(d) C_7-C_8 (0.198)-(0.801)

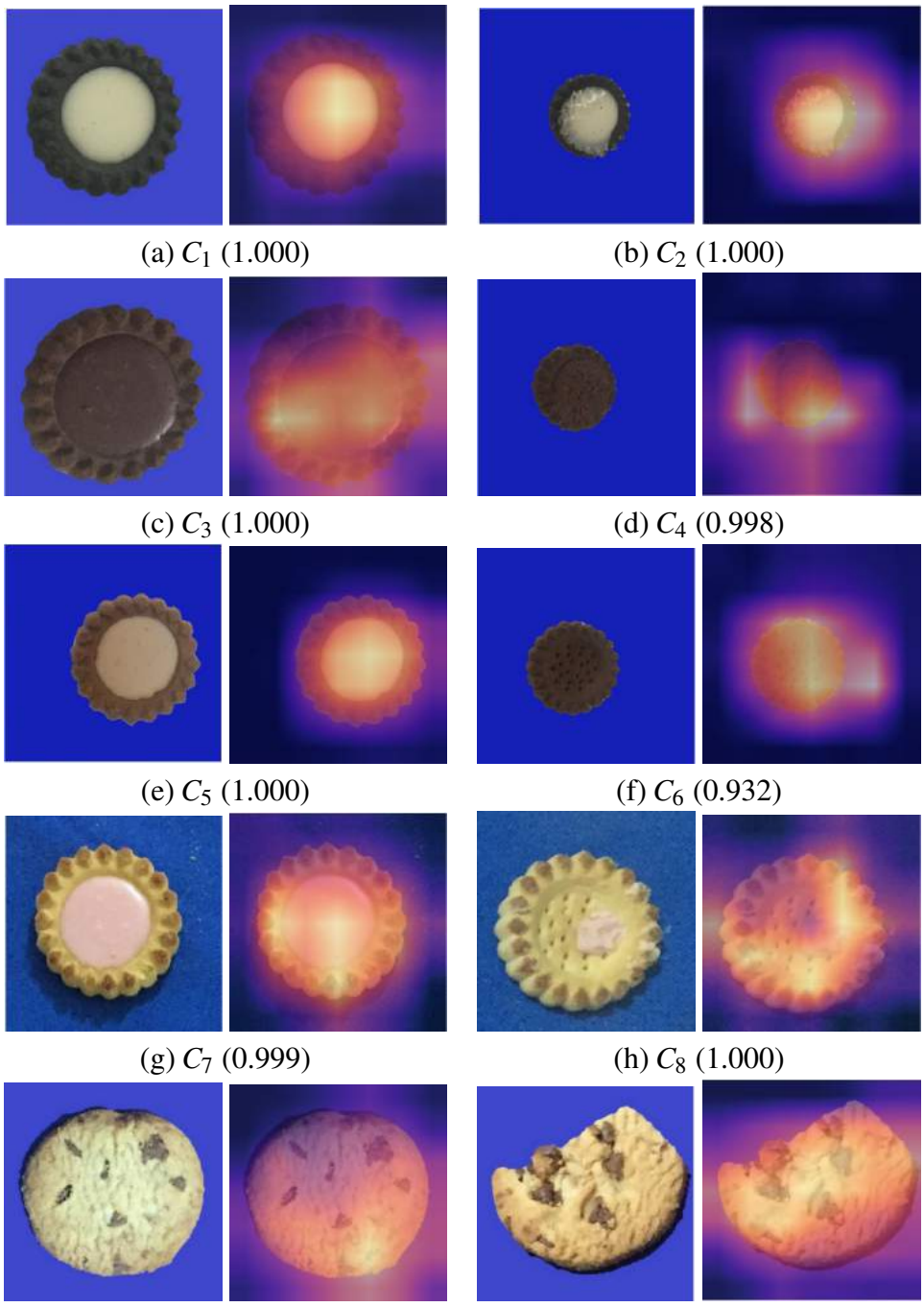


(e) C_8-C_7 (0.117)-(0.878)

(f) C_9-C_{10} (0.235)-(0.764)

(g) C_{10}-C_9 (0.351)-(0.649)

Figura 24: Exemplos de imagens, originais (à esquerda) e resultantes da análise do Grad-CAM (ao centro e à direita), mais confundidas obtidas pela arquitetura EfficientNetB3. As imagens ao centro e à direita representam os mapas de ativação referentes às classes verdadeiras e preditas, respectivamente. São indicadas as classes confundidas (no formato classe verdadeira seguida da classe predita), bem como suas respectivas probabilidades entre parênteses.



(a) C_1 (1.000) (b) C_2 (1.000)

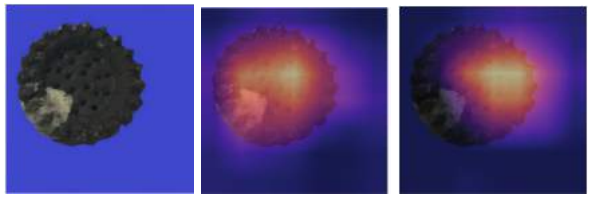
(c) C_3 (1.000) (d) C_4 (0.998)

(e) C_5 (1.000) (f) C_6 (0.932)

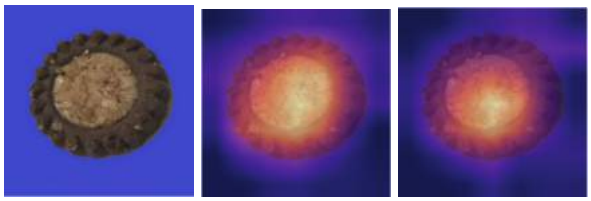
(g) C_7 (0.999) (h) C_8 (1.000)

(i) C_9 (0.992) (j) C_{10} (1.000)

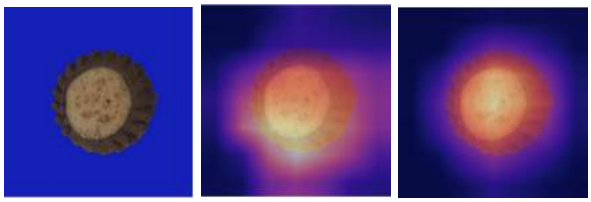
Figura 25: Exemplos de imagens, original (à esquerda) e resultante da análise do Grad-CAM (à direita), menos confundidas obtidas pela arquitetura ResNet50, para cada uma das classes C_1 a C_{10} (a)-(j), respectivamente. São apresentadas as classes preditas e suas respectivas probabilidades entre parênteses.



(a) C_2-C_6 (0.125)-(0.875)



(b) C_5-C_6 (0.496)-(0.504)



(c) C_6-C_5 (0.485)-(0.515)

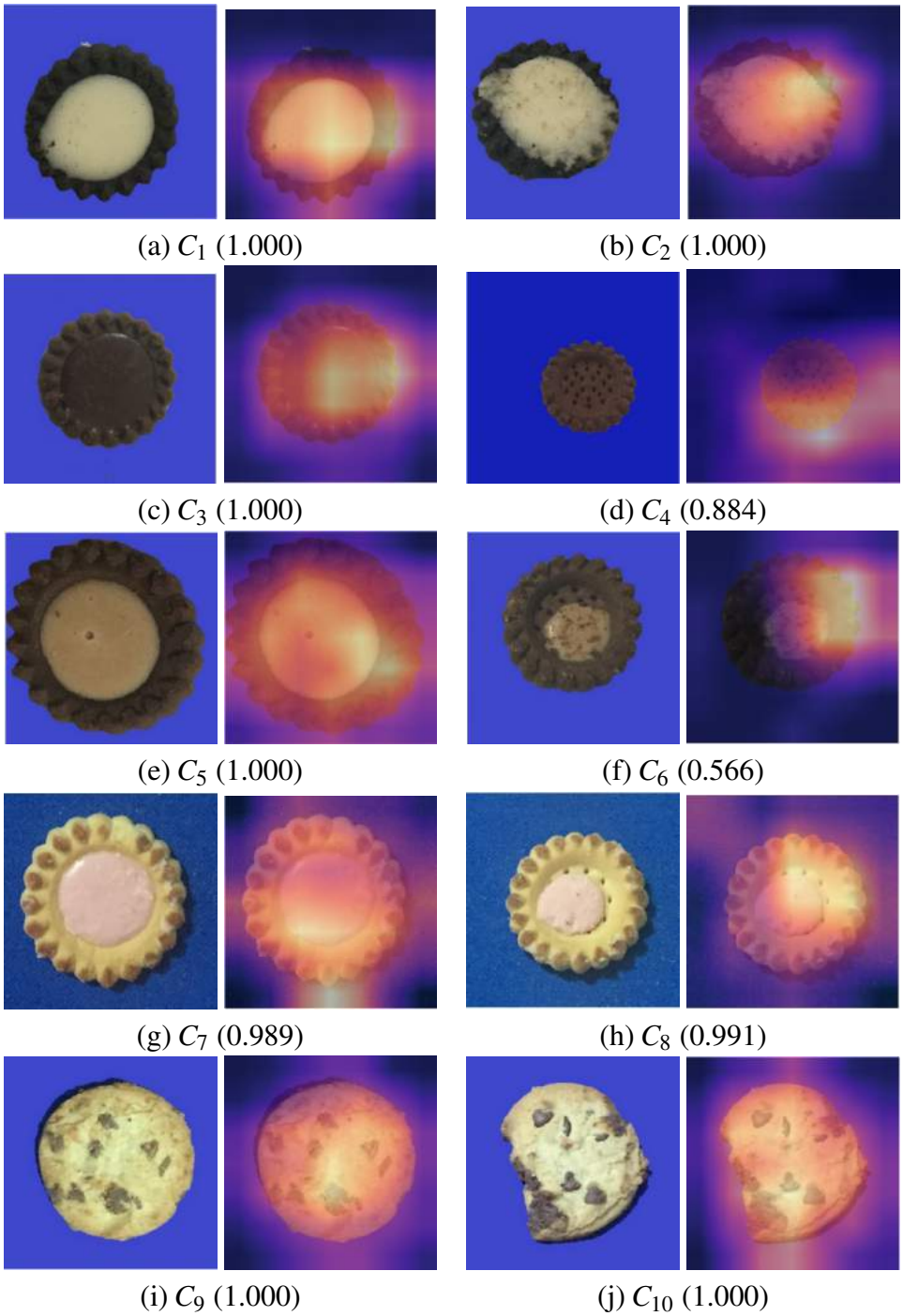
(d) C_7-C_8 (0.269)-(0.731)

(e) C_8-C_7 (0.036)-(0.964)

(f) C_9-C_{10} (0.006)-(0.994)

(g) C_{10}-C_9 (0.459)-(0.540)

Figura 26: Exemplos de imagens, originais (à esquerda) e resultantes da análise do Grad-CAM (ao centro e à direita), mais confundidas obtidas pela arquitetura ResNet50. As imagens ao centro e à direita representam os mapas de ativação referentes às classes verdadeiras e preditas, respectivamente. São indicadas as classes confundidas (no formato classe verdadeira seguida da classe predita), bem como suas respectivas probabilidades entre parênteses.



(a) C_1 (1.000) (b) C_2 (1.000)

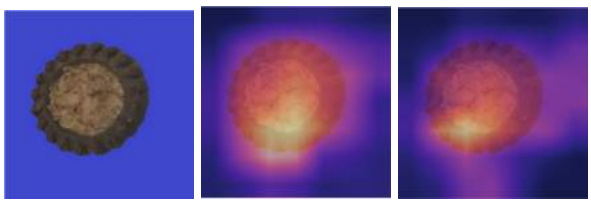
(c) C_3 (1.000) (d) C_4 (0.884)

(e) C_5 (1.000) (f) C_6 (0.566)

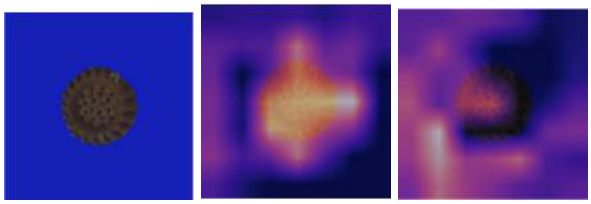
(g) C_7 (0.989) (h) C_8 (0.991)

(i) C_9 (1.000) (j) C_{10} (1.000)

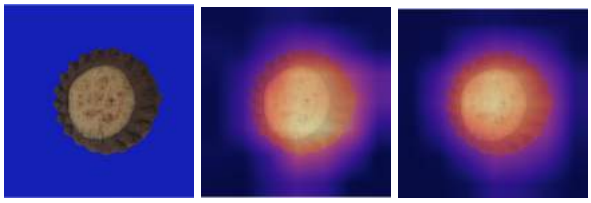
Figura 27: Exemplos de imagens, original (à esquerda) e resultante da análise do Grad-CAM (à direita), menos confundidas obtidas pela arquitetura VGG19, para cada uma das classes C_1 a C_{10} (a)-(j), respectivamente. São apresentadas as classes preditas e suas respectivas probabilidades entre parênteses.



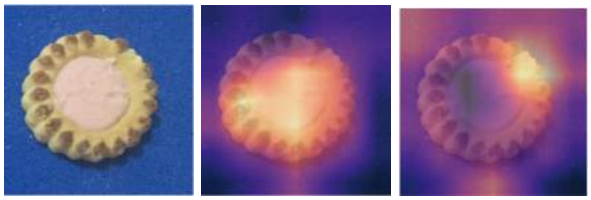
(a) C_5-C_6 (0.162)-(0.837)



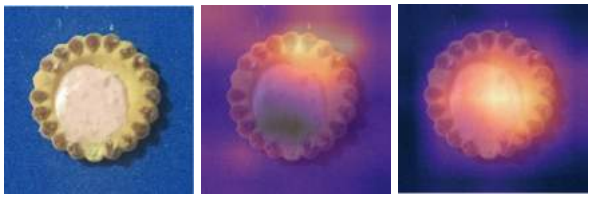
(b) C_6-C_4 (0.394)-(0.579)



(c) C_6-C_5 (0.367)-(0.633)



(d) C_7-C_8 (0.416)-(0.584)



(e) C_8-C_7 (0.243)-(0.757)

(f) C_9-C_{10} (0.426)-(0.561)

Figura 28: Exemplos de imagens, originais (à esquerda) e resultantes da análise do Grad-CAM (ao centro e à direita), mais confundidas obtidas pela arquitetura VGG19. As imagens ao centro e à direita representam os mapas de ativação referentes às classes verdadeiras e preditas, respectivamente. São indicadas as classes confundidas (no formato classe verdadeira seguida da classe predita), bem como suas respectivas probabilidades entre parênteses.