Relatório Final

Auxílio Regular FAPESP - Processo 2018/23392-5

Representações unificadas considerando atributos visuais e de semântica textual em tarefas de reconhecimento em imagens

Juliana M. Crivelli / Moacir A. Ponti

Instituto de Ciências Matemáticas e de Computação — Universidade de São Paulo

São Carlos, SP

Março de 2019 - Janeiro de 2020

Resumo

O sistema cognitivo-visual humano é capaz de abstrair conceitos visuais a partir de múltiplos elementos em uma cena. Por exemplo, é possível categorizar uma fotografia com pessoas no tema trabalho ou férias com base em atributos visuais como as roupas com as quais as pessoas estão vestidas, e os objetos na cena. Do ponto de vista de visão computacional e reconhecimento de padrões, essas representações poderiam ser identificadas como a mesma categoria. Assim, as características visuais abstratas comumente extraídas por métodos de visão computacional são comumente insuficientes, sendo necessário complementar com informação semântica. Nesse projeto serão investigadas informações semânticas complementares às características visuais abstratas. Em particular, utilizaremos características obtidas por redes neurais convolucionais como representações visuais abstratas, e as complementaremos com informação categórica textual a partir de métodos de reconhecimento de objetos ou anotações. Como resultado, primeiramente pretendemos entender o ganho nas representações quando as características são combinadas, e em segundo lugar como estender os métodos para traduzir características visuais em textual e vice-versa. Aplicações possíveis incluem a descrição de cenas, detecção de sub-categorias visuais, detecção de anomalias, entre outros.

Abstract

The human cognitive-visual system is capable of abstracting visual concepts from multiple elements in a scene. For instance, it is possible to categorise a photo with people in the theme work or vacation according to visual attributes such as clothing and objects in the scene. From the point of view of computer vision and pattern recognition, these representations could be identified as the same category. Therefore, abstract visual characteristics commonly extracted by computational vision methods are usually insufficient, being necessary complement with semantic information. In this project semantic information complimentary to abstract visual characteristics will be investigated. In particular, characteristics obtained by convolution neural networks will be used as abstract visual representations, complemented by categorical textual information from object recognition methods or annotations. As result, first we aim to understand improvement in representation when characteristics are combined, and in second place how to extend the methods for translating visual characteristics in textual characteristics and vice versa. Possible applications include scene description, visual sub-categories detection, anomaly detection and others.

Sumário

1 Introdução

O sistema cognitivo-visual humano é capaz de analisar e interpretar com precisão imagens complexas, de modo quase instantâneo, abstraindo conceitos visuais a partir de múltiplos elementos em uma cena. Por exemplo, é possível categorizar uma fotografia com pessoas no tema trabalho ou férias com base em atributos visuais como as roupas com as quais as pessoas estão vestidas, e os objetos na cena. Humanos também são capazes de gerar uma variedade de descrições textuais em cima da mesma cena, abordando diferentes aspectos observados. Em geral, humanos são capazes de abstrair conceitos a partir da visualização de poucos exemplos [Gonzalez and Woods, 2002].

Em contrapartida, a visão computacional possui maior limitação com relação à abstração e aprendizado de conceitos visuais. Por exemplo, em um problema de classificação como o anteriormente citado, as cenas poderiam ser rotuladas com a mesma categoria. Com a maior demanda por sistemas mais acurados e eficientes envolvendo imagens e vídeos, a áreas da computação relacionadas à visão e reconhecimento de padrões está em crescente expansão na última década, tanto pelo aumento significativo de dados e imagens armazenadas nos últimos anos quanto pelo aumento de poder computacional disponível [Szeliski, 2010].

Há diferentes tipos de características que podemos obter a partir do conteúdo visual, gerando representações abstratas ou ainda representações semânticas. As representações mais abstratas dizem respeito à um conjunto de características visuais de mais difícil interpretação como: cor, textura, orientação entre outros. Em contraste, as representações semânticas comumente incluem categorias de objetos, cenas ou ações e portanto estão mais próximas da linguagem humana. De forma ainda mais flexível, características textuais permitem maior liberdade para utilizar a semântica e interpretar a cena, gerando descrições ainda mais similares à percepção humana.

Métodos como Bag of Visual Words (BoVW) [Fei-Fei and Perona, 2005], Fisher Vectors [Perronnin et al., 2010] foram propostos em meados da década de 2000 e até próximo a 2010 com o objetivo de melhor codificar características visuais em uma cena, obtendo dicionários visuais com os quais relacionar características (de cor, textura, orientação, etc.) encontradas em imagens [Haralick et al., 1973]. Na década de 2010, em particular após a publicação dos métodos de redes neurais convolucionais (CNNs, do inglês Convolutional Neural Networks) AlexNet, VGGNet e Inception [Krizhevsky et al., 2012; Simonyan and Zisserman, 2014; Szegedy et al., 2015], essas passaram a dominar as soluções de visão computacional quando se trata da extração de características. Ainda que produzindo maior acurácia do que os métodos anteriores [Sharif Razavian et al., 2014], as características baseadas em CNNs são abstratas podendo gerar ambiguidade na interpretação de cenas

mais complexas, em particular quando considerados contextos, planos de fundo e posições dos objetos nas cenas [Ponti et al., 2017], e prejudicando a performance de sistemas com poucos dados anotados [Fu et al., 2014]. Assim, esse projeto tem como objetivo investigar o aprendizado e combinação de representações unificadas, que considerem atributos visuais, semânticos e textuais obtidos a partir de bases de dados de imagens, e seu impacto nas tarefas de classificação e busca visual. Os objetivos mais específicos são:

1. Estudar e implementar a extração de características visuais e semânticas, bem como textuais;

2. Investigar o impacto da combinação dos atributos visuais, semânticos e textuais em tarefas de classificação e busca visual;

3. Desenvolver e avaliar métodos que sejam capazes de aprender uma representação unificada considerando os três tipos de atributos citados, avaliando sua qualidade nas tarefas de reconhecimento.

Este relatório visa apresentar os resultados obtidos no decorrer projeto. Como ponto de partida, é utilizada a representação de imagens produzida por CNNs pré-treinadas [Sharif Razavian et al., 2014] para a obtenção de características visuais abstratas e o espaço de características obtidas é avaliado através dos métodos de visualização t-SNE (t-distributed stochastic neighbor embedding) Maaten and Hinton [2008] e PCA (Principal Component Analysis). Em seguida, são apresentadas as análises e resultados da fusão de características visuais e semânticas, avaliados comparando o seu desempenho com um classificador SVM.

2 Conceitos fundamentais

2.1 Representações e características

Nesta seção são apresentados conceitos e técnicas que servirão como base para o trabalho proposto neste projeto. Há diferentes tipos de características que podemos obter a partir do conteúdo visual, gerando representações abstratas ou ainda representações semânticas. As representações mais abstratas dizem respeito à um conjunto de características visuais de mais difícil interpretação como: cor, textura, orientação entre outros. Em contraste, as representações semânticas comumente incluem categorias de objetos, cenas ou ações e portanto estão mais próximas da linguagem humana. De forma ainda mais flexível, características textuais permitem maior liberdade para utilizar a semântica e interpretar a cena, gerando descrições ainda mais similares à percepção humana.

2.1.1 Características visuais

É essencial definir as características que melhor representem conceitos. Os atributos mais comumente explorados são cor, textura e forma ou orientação dos objetos e regiões em uma imagem.

— **Cor**: modo de perceber e interpretar uma determinada frequência de luz. O espectro de luz visível aos humanos corresponde a faixa de 400 a 700nm e suas cores podem ser especificadas como a combinação em diferentes proporções de vermelho, verde e azul (Modelo RGB) Dentre os métodos para descrição de cor, destacamos: **BIC (Border / Interior Classification)**: utiliza uma versão quantizada (8 valores) da imagem para computar dois histogramas, um para píxeis classificados como interior e outro para píxeis classificados como borda. Um píxel é classificado como borda se ao menos um de seus vizinhos tiver uma cor quantizada diferente e é classificado como interior no caso oposto [Stehling et al., 2002].

— **Textura**: descreve a imagem em termos de sua suavidade, rugosidade e regularidade. As principais abordagens entre os descritores de texturas partem de medidas estatísticas, técnicas estruturais e baseadas no espectro de Fourier. Dentre os métodos para descrição, destacamos os Descritores de Haralick et al. [1973] obtidos a partir de matrizes de co-ocorrência:

- 6 características de textura são obtidas a partir das matrizes: probabilidade máxima, correlação, contraste, energia, homogeneidade e entropia [Haralick et al., 1973].

— **Orientação**: características que tentam capturar a forma dos objetos por meio das orientações das principais regiões e bordas. O método de Histograma de Orientação do Gradiente (HOG) Dalal and Triggs [2005] tem destaque nesse sentido na última década.

— **Características *CNN off-the-shelf***: ao invés de selecionar um subconjunto de caracerísticas específico, utiliza uma rede neural convolucional pré-treinada como extrator de características. Serão explorados modelos como VGGNet-16, Inception e MobileNet [Ponti et al., 2017], para os quais se fornece como entrada uma imagens e as características são as saídas de alguma das camadas da rede neural.

2.1.2 Atributos semânticos

Em contraste com os atributos visuais, os quais são valores obtidos a partir dos pixels e suas relações na imagem, os atributos semânticos se referem ao significado simbólico dos elementos na imagem, ou ainda a interpretação de uma cena.

Em aplicações de aprendizado de máquina que lidam com classificação, a classe obtida para cada input pode ser um atributo semântico, associando a imagem a uma categoria entre as pré-estabelecidas na aplicação. Detecção de objetos é uma tarefa geralmente associada a classificação, fornecendo por exemplo uma bounding box e uma categoria por objeto [Redmon et al., 2015].

2.1.3 Atributos textuais

É possível estabelecer uma relação entre o conteúdo visual e sua semântica e uma descrição textual que um humano pode produzir ao anotar esse conteúdo. Um exemplo ocorre em recuperação de imagens baseada em conteúdo: nesse cenário os atributos textuais formam uma string de busca pelas imagens desejadas Wang and Zhang [2009]. Outro trabalho usa uma descrição textual das características visuais Tsai et al. [2012], com a vantagem de diminuir o tamanho da representação mantendo a performance do sistema.

2.2 Aprendizado de Máquina

Aprendizado de máquina é um campo da ciência da computação contido dentro da área de inteligência artificial, utilizando técnicas matemáticas e estatísticas, de teoria da informação e até mesmo inspiradas na biologia (como as redes neurais) para criar programas capazes de melhorar sua performance em uma tarefa através de experiência [Mitchell, 1997].

Podemos dizer que neste contexto, o aprendizado se caracteriza por uma função ou mapeamento aprendida a partir de exemplos. Por exemplo inferir $f : X \to Y$, em que X é o espaço de imagem (ou representação das imagens), e Y é o espaço dos rótulos (ou classes, categorias) a qual a imagem pertence. Ou $f : X \to \hat{X}$, em que $\hat{X}$ é uma reconstrução ou versão processada da imagem X.

A performance, o quanto a função aprendida se aproxima da função ideal para a aplicação deve ser medida através de uma métrica adequado ao tipo de tarefa. Para a classificação (designar uma classe para um exemplo dado), por exemplo, é comum utilizar medidas baseadas em acurácia, que representam a taxa de exemplos classificados corretamente no total de exemplos apresentados para teste.

Neste trabalho utilizaremos redes neurais, um paradigma de aprendizado de máquina que utiliza unidades simples, chamadas de neurônios, compondo camadas que se conectam de modo a obter f. Apesar da dificuldade de interpretação humana dos pesos e resultados intermediários obtidos em uma rede neural, elas tem se mostrado eficazes para problemas de classificação de imagem [Ponti et al., 2017]. Entre os conceitos essenciais para o entendimento de redes neurais e dos métodos utilizados neste trabalho podemos destacar Goodfellow et al. [2016]:

Neurônio: unidade básica de uma camada densa ou convolucional em uma rede neural. Realiza uma combinação linear de seus valores de entrada, produzindo uma única saída. Cada neurônio possui um conjunto de pesos, que determinam a importância de uma determinada entrada para a saída daquele neurônio.

Camada densa: todos os neurônios de uma camada densa (também conhecida como *fully connected* ou FC) estão conectados a todos os neurônios da camada anterior. Deste modo, as entradas de um neurônio são todas as saídas da camada prévia.

Camada convolucional: recebe uma imagem como entrada e cada neurônio da camada aplica nela um filtro em uma operação de convolução. O filtro atua como uma matriz de pesos em cima de um pixel e seus vizinhos e é o mesmo para toda a imagem naquele neurônio, gerando $\hat{X}$.

Deep learning: permite aprender representações hierárquicas por meio da composição de camadas [Ponti et al., 2017]. Assim, ao invés de aprender uma única função que mapeia o espaço de entrada no espaço objetivo, métodos profundos visam aprender uma série de funções aninhadas que comumente representam as L camadas de redes neurais:

$$f_1 : X \rightarrow X_1 \tag{1}$$

$$f_2 : X_1 \rightarrow X_2 \tag{2}$$

$$\cdots \tag{3}$$

$$f_L : X_{L-1} \rightarrow X_L \tag{4}$$

É interessante notar que as representações intermediárias, principalmente no caso de camadas convolucionais, podem ser utilizadas para extração de características (selecionadas de acordo com os filtros) que se tornam mais abstratas conforme a proximidade da camada em relação a saída final da rede.

3 Metodologia

3.1 Visão geral do método

A Figura 1 exibe a sequência de procedimentos do uso de diferentes atributos para o aprendizado de representações. Esses atributos podem ser combinados ou selecionados de maneira a melhorar sistemas de reconhecimento, como por exemplo a classificação ou a busca visual. Assim, esse projeto irá implementar métodos que sejam capazes de extrair os atributos (1, 2 e 3), posteriormente aprender uma combinação desses atributos em uma representação unificada (4) e finalmente avaliar essa representação em cenários de reconhecimento (5). No projeto original, previmos a utilização de características textuais, porém a obtenção dessas características se mostrou um problema de pesquisa por si só, requerendo o estudo de representações ou mapeamentos do tipo Word2Vec. Assim, optamos por substituir as características textuais por atributos espaciais, que codificam a localização do objeto descrito no atributo semântico.

Os pontos de partida para a investigação de cada uma das etapas do pipeline serão descritos na seção seguinte.

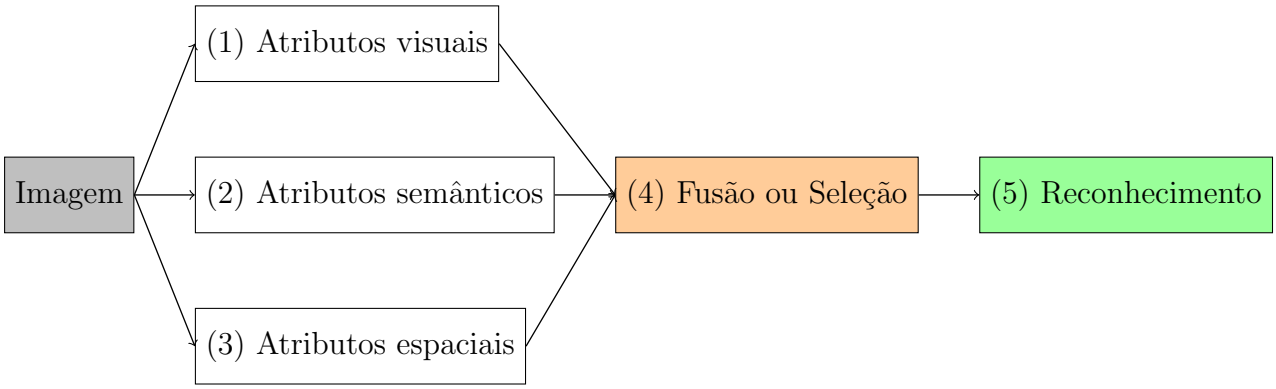

Figura 1: Sequência completa dos procedimentos para investigar diferentes atributos, combinar e realizar reconhecimento sobre o aprendizado realizado.

3.2 Investigação e implementação

- **Extração de caracterísicas visuais**: o principal método investigado será a extração de espaços de características por meio do uso de redes neurais do tipo CNN pré-treinadas [Sharif Razavian et al., 2014] ou com ajuste fino quando a base de dados tiver exemplos rotulados disponíveis.

- **Extração de caracterísicas semânticas**: utilizando as anotações existentes na base de dados (na forma de oráculo), ou por meio de redes neurais que detectam e localizam objetos, como YOLO [Redmon et al., 2015].

- **Extração de caracterísicas espaciais**: utilizando as anotações existentes na base de dados (na forma de oráculo).

- **Combinação das representações**: como forma de comparar o espaço latente gerado, aplicaremos ainda combinações das representações Ponti Jr [2011] por meio de fusão por concatenação (*early fusion*): em que os espaços serão concatenados sem realização de qualquer aprendizado adicional.

3.3 Avaliação

As representações geradas e suas combinações serão avaliadas medindo a sua eficiência em tarefas de reconhecimento, em particular em problemas de classificação e busca visual. As métricas de avaliação de cada tarefa podem ser usadas como maneira de medir a qualidade das representações geradas.

Para classificação, a escolha é o método Support Vector Machines (SVM) Cortes and Vapnik [1995], que possui garantias de aprendizado de um hiperplano linear de separação das classes do problema, o que permite avaliar o espaço de características em termos da sua capacidade discriminativa.

4 Desenvolvimento

4.1 Análise de coerência do espaço latente

4.1.1 Bases de Dados

Para o propósito de estudos iniciais foi utilizada a base de dados **CIFAR-10**, pois é simples o suficiente para prototipagem rápida dos experimentos mas ainda assim possui uma certa complexidade que exige maior atenção à arquitetura da rede e treinamento para obter maior acurácia. A base contém imagens pertencentes a dez classes: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck (avião, automóvel, pássaro, gato, cervo, cachorro, sapo, cavalo, navio e caminhão). A base consiste em 6000 imagens coloridas por classe, de 32 por 32 pixels. Os grupos de treinamento e teste possuem respectivamente 50000 e 10000 imagens. A Figura 11 possui exemplos de imagens presentes na base de dados.

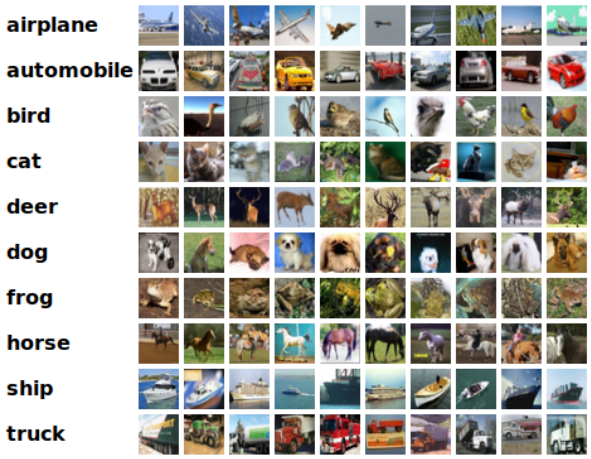

Figura 2: Exemplos de images da base de dados CIFAR-10

4.1.2 Extração de características

Nesta etapa do projeto buscou-se trabalhar com características visuais e semânticas. O método investigado para a obtenção de características visuais foi a extração de espaços de características por meio do uso de redes neurais do tipo CNN pré-treinadas [Sharif Razavian et al., 2014] ou com ajuste fino quando a base de dados tiver exemplos rotulados disponíveis. Para a extração de características semânticas foram utilizadas as anotações existentes na base de dados (na forma de oráculo).

Foram testadas 2 arquiteturas de redes: a primeira possui arquitetura arbitrária, treinadas rápido o suficiente para obter um direcionamento dos experimentos, porém com baixa acurácia. A outra rede tem uma arquitetura mais complexa e foi treinada utilizando uma quantidade maior de dados através de *data augmentation* e por um número maior de épocas.

4.1.3 Métodos de Visualização

O espaço de características gerado é multidimensional. Deste modo é necessário utilizar um método de redução da dimensionalidade para possibilitar a visualização. É desejado que o número de variáveis do conjunto de dados seja reduzido preservado ao máximo a informação contida nele. Deste modo, buscou-se aplicar dois métodos bastante conhecidos na literatura, descritos a seguir.

PCA (*principal component analysis*)

O método PCA funciona identificando quais componentes são mais relevantes para a informação e reorganizando-a de modo a apresentar uma representação mais compacta, que apesar de possuir uma certa perda de precisão ainda é capaz de reconstituir a informação original com um alto grau de fidelidade.

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	(None, 32, 32, 3)	0
conv2d_1 (Conv2D)	(None, 28, 28, 16)	1216
conv2d_2 (Conv2D)	(None, 24, 24, 16)	6416
conv2d_3 (Conv2D)	(None, 22, 22, 32)	4640
conv2d_4 (Conv2D)	(None, 20, 20, 32)	9248
conv2d_5 (Conv2D)	(None, 18, 18, 32)	9248
conv2d_6 (Conv2D)	(None, 18, 18, 64)	2112
flatten_1 (Flatten)	(None, 20736)	0
dense_1 (Dense)	(None, 128)	2654336
dense_2 (Dense)	(None, 128)	16512
dense_3 (Dense)	(None, 10)	1290

Total params: 2,705,018
Trainable params: 2,705,018
Non-trainable params: 0

Figura 3: Sumário da rede 1

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 32, 32, 32)	896
activation_1 (Activation)	(None, 32, 32, 32)	0
batch_normalization_1 (Batch	(None, 32, 32, 32)	128
conv2d_2 (Conv2D)	(None, 32, 32, 32)	9248
activation_2 (Activation)	(None, 32, 32, 32)	0
batch_normalization_2 (Batch	(None, 32, 32, 32)	128
max_pooling2d_1 (MaxPooling2	(None, 16, 16, 32)	0
dropout_1 (Dropout)	(None, 16, 16, 32)	0
conv2d_3 (Conv2D)	(None, 16, 16, 64)	18496
activation_3 (Activation)	(None, 16, 16, 64)	0
batch_normalization_3 (Batch	(None, 16, 16, 64)	256
conv2d_4 (Conv2D)	(None, 16, 16, 64)	36928
activation_4 (Activation)	(None, 16, 16, 64)	0
batch_normalization_4 (Batch	(None, 16, 16, 64)	256
max_pooling2d_2 (MaxPooling2	(None, 8, 8, 64)	0
dropout_2 (Dropout)	(None, 8, 8, 64)	0
conv2d_5 (Conv2D)	(None, 8, 8, 128)	73856
activation_5 (Activation)	(None, 8, 8, 128)	0
batch_normalization_5 (Batch	(None, 8, 8, 128)	512
conv2d_6 (Conv2D)	(None, 8, 8, 128)	147584
activation_6 (Activation)	(None, 8, 8, 128)	0
batch_normalization_6 (Batch	(None, 8, 8, 128)	512
max_pooling2d_3 (MaxPooling2	(None, 4, 4, 128)	0
dropout_3 (Dropout)	(None, 4, 4, 128)	0
flatten_1 (Flatten)	(None, 2048)	0
dense_1 (Dense)	(None, 10)	20490

Total params: 309,290
Trainable params: 308,394
Non-trainable params: 896

Figura 4: Sumário da rede 2

Geralmente é necessária a normalização das variáveis para que uma não tenha predominância sobre outra. No caso específico deste projeto, o vetor de características extraído das camadas intermediárias da rede já apresenta valores dentro da mesma faixa. Na próxima etapa busca-se identificar variáveis altamente correlacionadas, que podem apresentar informações redundantes e portanto podem ser simplificadas. Para isso é preciso computar a matriz de covariância e seus autovalores e autovetores.

Novas váriaveis (denominadas componentes principais) são geradas através de combinações lineares entre os autovetores e os dados originais (ambos transpostos). As componentes principais são ordenadas de tal forma que a primeira representa a maior quan-

tidade de informação possível a se codificar com uma variável, maximizando a variância entre as variáveis originais projetadas em um determinado eixo. A quantidade de informação decai conforme nos aproximamos da componente principal n, para um conjunto original de n variáveis. Isso decorre da ordenação dos autovetores de modo que seus autovalores (que representam o quanto aquele autovetor é relevante) fiquem em ordem decrescente. Podemos compor nosso novo espaço utilizando as duas primeiras componentes principais de modo a criar uma visualização 2D.

T-SNE (*t-Distributed Stochastic Neighbor Embedding*) Enquanto o PCA busca manter pontos distantes na geometria original distantes na geometria gerada, preocupando-se com relações globais, o método t-SNE possui uma abordagem local e não linear, preservando agrupamentos mas não necessariamente a relação entre os demais agrupamentos no espaço. O método converte as distâncias euclidianas entre cada par para uma medida probabilística de similaridade baseada na projeção da distância em uma distribuição normal, . Pontos atrelados aos dados originais são gerados em uma dimensão menor, com posicionamento aleatório. A cada passo o algoritmo busca aproximar as distâncias entre os pontos gerados para as distâncias medidas no espaço original ao minimizar uma função de custo através do gradiente descendente. Para prevenir alta densidade de pontos em uma única região na visualização em menor dimensão é utilizada a distribuição t de Student no cálculo das similaridades para os pontos no espaço desejado.

PCA + T-SNE Computacionalmente, o método t-SNE é bastante custoso. Para a realização das análises cada método foi testado individualmente e em uma combinação. O PCA foi utilizado primeiramente para trazer os vetores de características para dimensões mais gerenciáveis pelo t-SNE, que foi responsável pela segunda fase de redução de dimensionalidade. O processo diminuiu o tempo necessário para gerar a visualização do espaço 2-D.

4.1.4 Análise

Como ponto de partida, as redes neurais anteriormente descritas foram utilizadas para realizar um estudo acerca dos métodos de visualização e do comportamento dos exemplos apresentados à rede através de suas camadas intermediárias. Para facilitar a distinção visual entre as classes nos gráficos gerados os casos de teste foram restringidos a quatro classes. As redes foram treinadas e testadas com a mesma base de dados (CIFAR-10).

Os testes foram realizados utilizando ambas as redes, mas a Rede 1, apesar de treinada por menos tempo e mais simples apresentou resultados mais visivelmente separáveis em análises com os métodos acima explicados. O t-SNE apresentou uma maior capacidade

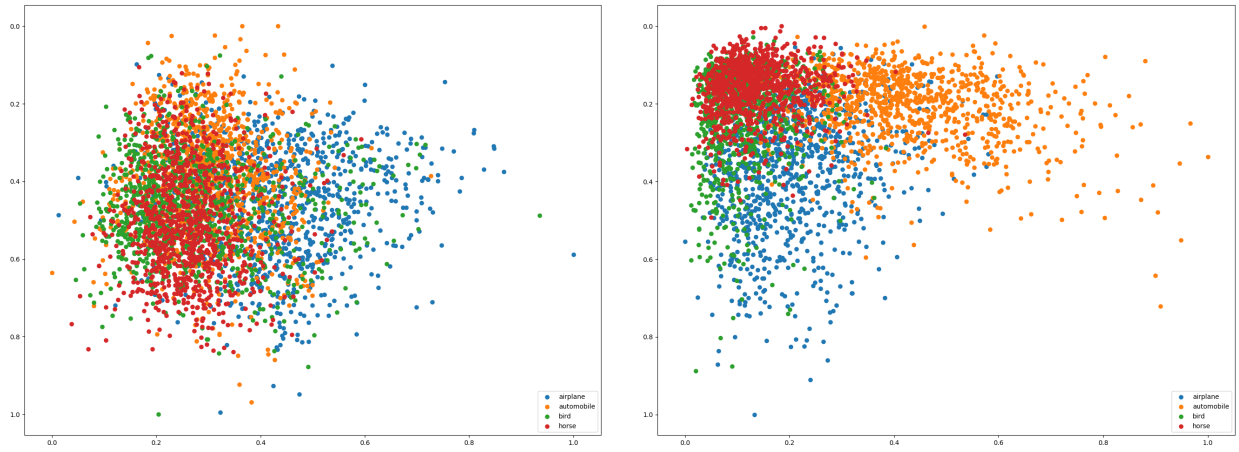

Figura 5: Utilizando a Rede 1 e o método PCA (a) 1ª camada convolucional (b) Última camada densa

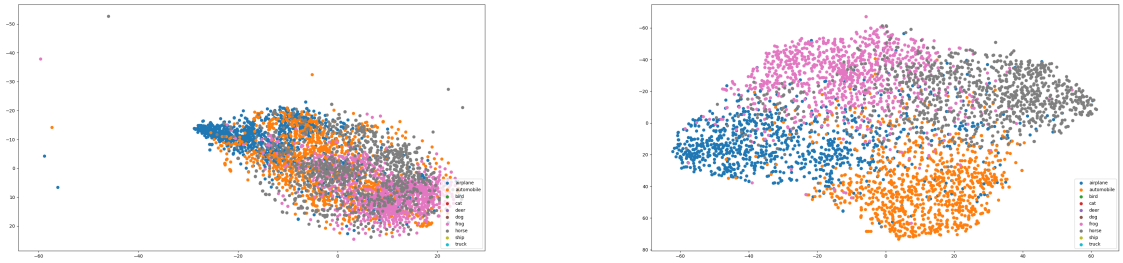

Figura 6: Utilizando a Rede 1 e o método t-SNE (a) 1ª camada convolucional (b) Última camada densa

de agrupar os exemplos teste de acordo com suas classes, com menos sobreposição de espaços. O resultado da combinação dos métodos foi muito semelhante ao t-SNE puro, porém foi executado com maior velocidade.

No experimento seguinte foi criado um gerador de imagens aleatórias que utiliza formas geométricas poligonais, retas e curvas de bézier de acordo com esquemas de cores para criar exemplos abstratos, que a princípio não pertencem a classe alguma. A Figura 7 apresenta alguns dos exemplos gerados.

Para essa etapa as redes treinadas em um momento anterior foram abandonadas em prol de utilizar a rede pré-treinada VGG-16, gerando previsões de classes pertencentes à ImageNet para os exemplos aleatórios. A fim de comparação, apenas resultados de imagens pertencentes a classes obtidas por exemplos com mais de 45% de confiança em suas previsões foram inseridos nos gráficos gerados. É possível notar que há uma separação clara entre imagens que pertencem ao conjunto de exemplos reais (independente da classe) e imagens geradas aleatoriamente, conforme a Figura 8.

Os exemplos aleatórios possuem dispersão maior no espaço e somente nos últimos

Figura 7: Exemplos de imagens aleatórias geradas

estágios da rede se aproximam dos seus respectivos representantes entre os exemplos reais,. Mesmo assim, ainda estão distribuídos em uma área de raio consideravelmente maior que os exemplos reais e os exemplos aleatórios aos quais foi atribuida a classe "macaw"estão espalhados por todo o espaço projetado, levantando questionamentos acerca do motivo de suas classificações com grau de confiança aceitável. É necessária maior investigação para descobrir se trata-se da rede de fato reconhecer um objeto em específico ou os resultados de exemplos retratados pelo gráfico são puramente aleatórios.

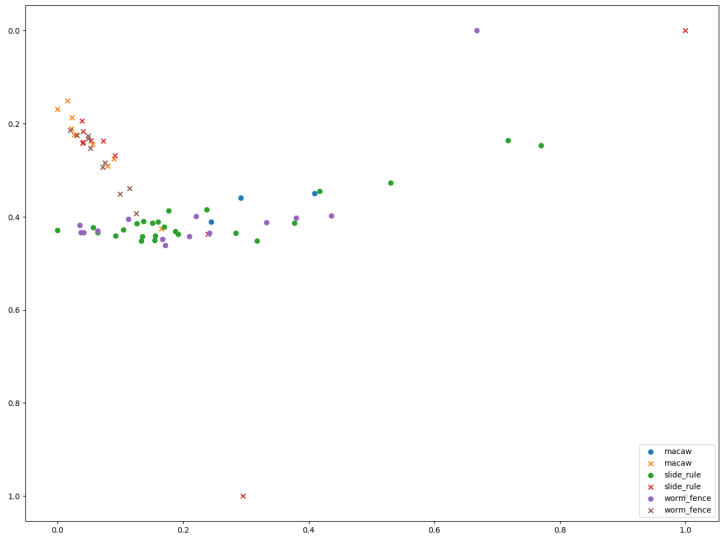

Figura 8: Exemplos aleatórios (o) e reais (x) em um bloco intermediário de camadas convolucionais da VGG-16

4.1.5 Detecção de Exemplos Artificiais

Utilizamos um classificador do tipo SVM (Support Vector Machines) linear para investigar se o espaço de características obtido na classificação de imagens aleatórias é compatível com o da classificação de imagens reais.

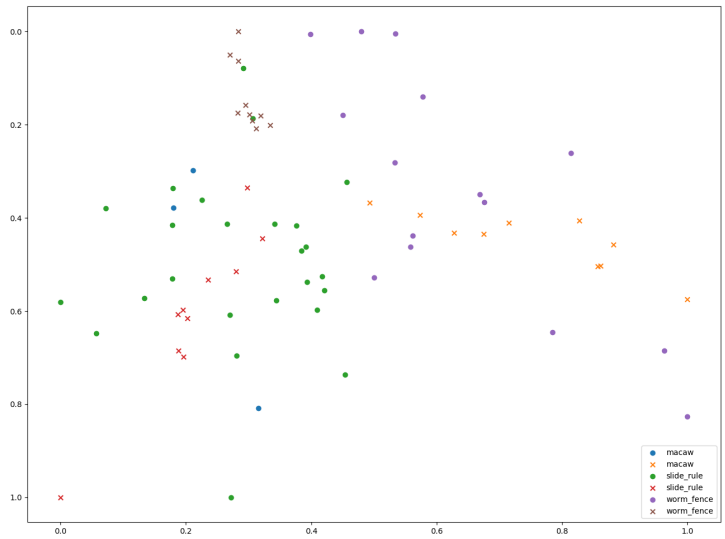

Figura 9: Exemplos aleatórios (o) e reais (x) no último bloco fully connected da VGG-16

Treinamos dois classificadores SVM, um utilizando a penúltima camada da VGG-16 em previsões de imagens reais pertencentes a base CIFAR-10 (denominado R) e outro igualmente utilizando penúltima camada da rede, porém executando previsões de imagens aleatórias (denominado A).

A fase de teste do classificador foi realizada utilizando imagens do tipo oposto. No caso da existência da proximidade entre os dois espaços de características é esperado que o classificador R preveja corretamente as classes de imagens aleatórias (utilizamos como parâmetro de comparação as classes previstas pela VGG-16). Analogamente para o classificador A.

O experimento foi realizado utilizando 50 exemplos (tanto reais quanto aleatórios) para as 10 classes mais frequentes entre aquelas previstas para imagens aleatórias e previstas com probabilidade de pertencer à classe superior a 0,1 (levando em conta que para as 1000 classes da Imagenet o limiar de previsão aleatória é 0,001).

Os resultados de acurácia obtidos foram 0,294 para o classificador R, isto é, 29,4% das vezes as imagens aleatórias utilizadas no teste obtiveram a mesma classe mais provável prevista pela VGG-16. Para o classificador A, a acurácia foi de 0,01, indicando que as classes das imagens reais testadas coincidiram com a maior previsão de classe em apenas 1% das imagens apresentadas ao classificador.

Em ambos os casos as classes previstas apresentaram poucas variações, quase sempre prevendo a mesma classe X para o classificador R e a mesma classe Y para o classificador A. Isto evidencia como o espaço de características não tem proximidade, apesar da alta acurácia prevista pela rede. Assim, como a rede neural possui um classificador (codificado na última camada) com alta capacidade em termos do espaço de funções admissíveis, o resultado de probabilidade atribuído a um exemplo artificial não corresponde ao conceito

da classe aprendida, e pode levar a interpretações incorretas. Nosso resultado oferece uma nova forma de detectar exemplos artificiais (possivelmente adversariais) oferecidos para a rede, por meio da análise do espaço de características na camada anterior à da predição.

4.2 Fusão de atributos

4.2.1 PASCAL VOC

O dataset PASCAL Visual Object Classes (VOC) contém em um de seus subsets imagens de uma ou mais pessoas suas respectivas bounding boxes e a ação que elas estão realizando, por exemplo tocar um instrumento musical, caminhar ou usar o computador. São ao total 10 classes de ações e uma classe de outros, que abrange casos onde foram detectadas pessoas porém as ações realizadas não se encaixam nas 10 categorias.

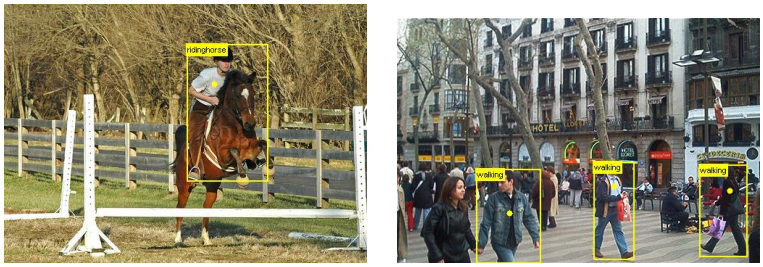

Figura 10: Exemplos do dataset PASCAL VOC para detecção de pessoas e ações

Hipotisamos que informações relativas ao contexto auxiliariam a prever a ação, mesmo que o objetivo final seja apenas classificar a ação realizada. Por esta razão, além do treinamento que obtém características visuais através de CNNs investigaremos outras características semânticas das imagens utilizando a rede YOLO.

4.2.2 YOLO + MS COCO

A rede YOLO v3 (You Only Look Once) fornece bounding box de objetos detectados, prevendo por regressão logística a probabilidade de existir um objeto no bounding box, em oposição a algum outro elemento como grama ou água. Os labels atribuídos ao bounding box são multiclasses, o que melhora a classificação no caso de domínios com classes com interseções (por exemplo uma base que contenha as classes pessoa e mulher) [Redmon et al., 2015].

Utilizaremos a YOLO treinada no dataset Common Objects in Context (COCO), que contém imagens de cenas complexas do dia-a-dia, com um ou mais objetos em seus contextos naturais, sem se fixar a uma vista e posicionamento ditos icônicos (Lin et al.

[2014]). A base possui 91 tipos de objetos, somando 2,5 milhões de instâncias anotadas dentro de 328 mil imagens.

Cada imagem possui atreladas a ela as categorias de objetos presentes, suas instâncias segmentadas e um conjunto de 5 legendas textuais. Um exemplo de imagem presente da base de dados é mostrado na Figura 11.

Atributos textuais
a person standing by some flowers and a rail.
a man is standing near a large field taking a photo
man taking a picture of a scene
a man in a green shirt taking a picture.
a man that has green shirt looking over a rail

Figura 11: (a) Imagem original (b) Objetos segmentados (person, umbrella, handbag, chair, potted plant, cell phone)

4.2.3 Análise

Para analisar se a fusão de características visuais semânticas melhora a performance em tarefas de classificação realizamos uma comparação entre os resultados obtidos pelas seguintes redes e classificadores na tarefa de prever ações anotadas na base VOC:

1. CNN pré-treinada com a Imagenet e SVM treinado com o espaço latente (penúltima camada) de exemplos da base VOC.

2. Rede YOLOv3 pré-treinada na MS COCO e SVM treinado com características extraídas pela rede em cima de exemplos da base VOC.

3. CNN pré-treinada com a Imagenet, rede YOLOv3 pré-treinada na MS COCO e SVM treinado com concatenação direta do espaço latente da primeira rede com as características extraídas pela segunda rede.

Como forma de acelerar a execução dos experimentos e suas variações, já que carregar em memória mais de 6 mil imagens coloridas de tamanhos variados é uma tarefa demorada e que exige uma quantidade considerável de memória, a geração de vetores

de características em espaços latentes foi separada da análise de suas performances. Os vetores obtidos foram divididos em conjuntos de treinameto e teste na proporção 2:1.

No caso da CNN pré-treinada optou-se pela Resnet50 devido a sua penúltima camada mais compacta em relação às demais opções (tamanho 2048). Após a predição dos exemplos da VOC a classe de maior probabilidade foi salva, assim como os vetores latentes gerados. O resultado foi compactado utilizando PCA para 100 dimensões por cada exemplo.

No caso da YOLO, as previsões de bounding box de objetos foram consideradas apenas se atingissem o valor mínimo de 50% de confiança e estivessem entre os 10 primeiros objetos detectados. O vetor gerado para cada imagem contêm as coordenadas x e y máximas e mínimas e o número da classe prevista para cada objeto detectado acima do limite. Caso a imagem não atinja 10 imagens que atendem as condições, o restante do espaço é preenchido com o vetor [0, 0, 0, 0, 80], representando coordenadas vazias e uma classe não existente. Deste modo, conseguimos garantir a padronização de tamanho dos vetores semânticos. O tamanho total da representação é de 150 componentes, uma compactação significativa.

Existem diversos métodos para realizar a fusão de características, por exemplo a utilização de autoencoder [Cavallari et al., 2018]. Neste trabalho foi explorada a fusão por concatenação simples, ou seja, o vetor de características no espaço latente foi extendido e as informações semânticas de detecção de objetos e textuais fornecidas pelas anotações de classe do objeto (codificadas numericamente) foram colocadas ao lado dos dados já existentes. O conjunto das três informações passou a ser considerado um único vetor para propósitos de treinamento e teste. Um cuidado necessário foi de realizar operações de normalização para que os valores de um tipo de característica não predominasse em relação a outra.

Na tabela 1 podemos observar os resultados obtidos ao executar o classificador SVM treinado nas condições descritas acima e utilizando os dados processados e particionados conforme descrito anteriormente. Os experimentos foram realizados 10 vezes e o resultado registrado na tabela é a média aritmética dos valores obtidos. Através da comparação dos experimentos é possível notar que há uma leve melhoria na tarefa de classificação ao utilizar não apenas informações visuais mas também semânticas e textuais. Salienta-se que o conjunto final de vetores gerado é mais compacto do que o obtido através de uma utilização padrão de uma CNN pré-treinada. Futuros experimentos podem caminhar na direção de explorar novos métodos de fusão, extração e seleção das características semânticas e textuais para aprimorar classificação e outras tarefas, tais como a busca visual.

Tabela 1: Métricas obtidas nos experimentos 1 a 3

	Train Score	Test Score	Accuracy	Precision	Recall
1 - ResNet50	0.8038	0.5216	0.5216	0.5460	0.5103
2 - YOLOv3	0.3954	0.3854	0.3854	0.0147	0.0287
3 - ResNet50 + YOLOv3	0.8534	0.5634	0.5634	0.5659	0.5340

5 Conclusões

Este projeto apresentou duas contribuições principais à área de deep learning: a primeira no estudo das representações ao longo das redes CNNs e da sua análise com relação a exemplos reais e gerados artificialmente. A segunda, expande a primeira ao enriquecer a representação com características semânticas e espaciais adicionais às visuais, mostrando que há ganho de acurácia nessas características adicionais mesmo utilizando uma representação compacta.

O experimento utilizando imagens reais e imagens geradas artificialmente demostrou a importância da análise do espaço latente, observando camadas anteriores a preditiva como uma ferramenta extra para validar ou não os resultados apresentados pela rede. Mostrou-se ainda um promissor ponto de partida para um método de detecção de exemplos adversariais, onde uma rede neural deve ser capaz de detectar e separar exemplos falsos, gerados por uma outra rede pra se aproximar de integrantes plausíveis de um dataset.

No caso da fusão de características visuais, semânticas e espaciais um método simples como a concatenação foi capaz de apresentar melhoras nos resultados. Dando continuidade a essa linha, o método pode ser expandido com características adicionais semânticas utilizando codificação do tipo Word2Vec em descritores e anotações textuais. Outros métodos de fusão ainda podem ser explorados, tais como late fusion com o auxílio de autoencoders.

Referências

Cavallari, G. B., Ribeiro, L. S., and Ponti, M. A. Unsupervised representation learning using convolutional and stacked auto-encoders: a domain and cross-domain feature space analysis. In *SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI 2018)*, 2018.

Cortes, C. and Vapnik, V. Support vector machine. *Machine learning*, 20(3):273–297, 1995.

Dalal, N. and Triggs, B. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.

Fei-Fei, L. and Perona, P. A bayesian hierarchical model for learning natural scene categories. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 524–531. IEEE, 2005.

Fu, Y., Hospedales, T. M., Xiang, T., Fu, Z., and Gong, S. Transductive multi-view embedding for zero-shot recognition and annotation. In *European Conference on Computer Vision*, pages 584–599. Springer, 2014.

Gonzalez, R. C. and Woods, R. E. *Digital Image Processing*. Pearson/Prentice Hall, 2002.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. *Deep learning*, volume 1. MIT press Cambridge, 2016.

Haralick, R. M., Shanmugam, K., et al. Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, (6):610–621, 1973.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

Lin, T., Maire, M., Belongie, S. J., Bourdev, L. D., Girshick, R. B., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014. URL `http://arxiv.org/abs/1405.0312`.

Maaten, L. v. d. and Hinton, G. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

Mitchell, T. M. *Machine Learning*. McGraw-Hill, Inc., New York, NY, USA, 1 edition, 1997. ISBN 0070428077, 9780070428072.

Perronnin, F., Sánchez, J., and Mensink, T. Improving the fisher kernel for large-scale image classification. In *European conference on computer vision*, pages 143–156. Springer, 2010.

Ponti, M., Ribeiro, L. S., Nazare, T. S., Bui, T., and Collomosse, J. Everything you wanted to know about deep learning for computer vision but were afraid to ask. In *SIBGRAPI — Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T 2017)*, pages 1–26, 2017.

Ponti Jr, M. P. Combining classifiers: from the creation of ensembles to the decision fusion. In *Graphics, Patterns and Images Tutorials (SIBGRAPI-T), 2011 24th SIBGRAPI Conference on*, pages 1–10. IEEE, 2011.

Redmon, J., Divvala, S. K., Girshick, R. B., and Farhadi, A. You only look once: Unified, real-time object detection. *CoRR*, abs/1506.02640, 2015. URL `http://arxiv.org/abs/1506.02640`.

Sharif Razavian, A., Azizpour, H., Sullivan, J., and Carlsson, S. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 806–813, 2014.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Stehling, R. O., Nascimento, M. A., and Falcão, A. X. A compact and efficient image retrieval approach based on border/interior pixel classification. In *Proceedings of the eleventh international conference on Information and knowledge management*, pages 102–109. ACM, 2002.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

Szeliski, R. *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.

Tsai, S. S., Chen, H., Chen, D., Parameswaran, V., Grzeszczuk, R., and Girod, B. Visual text features for image matching. In *Multimedia (ISM), 2012 IEEE International Symposium on*, pages 408–412. IEEE, 2012.

Wang, X.-J. and Zhang, L. *Annotation-based Image Retrieval*, pages 85–88. Springer US, Boston, MA, 2009. ISBN 978-0-387-39940-9. doi: 10.1007/978-0-387-39940-9_17. URL `https://doi.org/10.1007/978-0-387-39940-9_17`.