

Detecção e monitoramento de vagas de estacionamento através de visão computacional

Vitor de Alencastro Lacerda - 11/0067142

Universidade de Brasília

November 13, 2016

Roteiro

- 1 Problema e motivação
- 2 Objetivos
- 3 Fundamentação Teórica
- 4 Redes Neurais Artificiais
- 5 Solução proposta
- 6 Resultados Experimentais
- 7 Conclusao

Roteiro

- 1 Problema e motivação**
- 2 Objetivos**
- 3 Fundamentação Teórica**
- 4 Redes Neurais Artificiais**
- 5 Solução proposta**
- 6 Resultados Experimentais**
- 7 Conclusao**

Problema

Problema

Procurar vagas em grandes estacionamentos é uma tarefa dispendiosa e que consome muito tempo.

Hipóteses

- As soluções para estacionamento fechados não são utilizadas nos estacionamentos abertos porque são caras, difíceis de instalar e de difícil escalabilidade.
- Utilizar algoritmos de visão computacional é uma solução barata, eficiente e eficaz para realizar esse monitoramento.
- Redes Neurais Artificiais são capazes de diferenciar entre vagas ocupadas e vagas livres.
- Detecção de movimento pode determinar quando um veículo estacionou em uma vaga.

Motivacao

Motivações para o trabalho:

- Financeira: Rondar estacionamentos em busca de vagas gasta tempo e dinheiro.
- Comercial: Sistema atrai clientes para estabelecimentos que o adotarem.

Roteiro

- 1 Problema e motivação
- 2 **Objetivos**
- 3 Fundamentação Teórica
- 4 Redes Neurais Artificiais
- 5 Solução proposta
- 6 Resultados Experimentais
- 7 Conclusao

Objetivos

Objetivo Geral

Desenvolver um sistema capaz de analisar imagens de uma câmera de vídeo para identificar vagas vazias e ocupadas em um estacionamento descoberto e ajudar motoristas a estacionar seus carros mais rapidamente.

Objetivos

Objetivos específicos:

- Mapear automaticamente as posições das vagas do estacionamento com mínima marcação de humanos.
- Informar aos motoristas a quantidade de vagas livres e ocupadas em uma região do estacionamento.
- Funcionar mesmo se a execução for iniciada em estacionamento ocupado.

Objetivos

Resultado Esperado

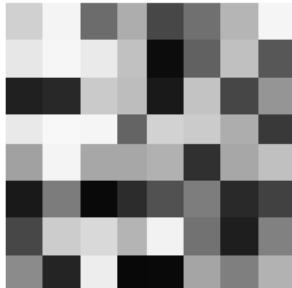
Um sistema barato e eficiente que seja capaz de facilitar a tarefa de encontrar vagas em grandes estacionamentos descobertos, sem o uso de sensores ou necessidade de grandes obras para instalação.

Roteiro

- 1 Problema e motivação
- 2 Objetivos
- 3 Fundamentação Teórica
- 4 Redes Neurais Artificiais
- 5 Solução proposta
- 6 Resultados Experimentais
- 7 Conclusao

Imagens em nível de cinza

- É necessário um modelo de representação de imagens em computadores.
- Forma mais simples são imagens de nível de cinza.
- Matriz $M \times N$ de elementos de 1bit que representam a luminosidade do pixel.



(a)

208	244	108	173	71	112	181	245
231	246	234	193	12	97	192	87
32	40	202	189	25	195	70	149
233	248	245	100	210	203	173	57
161	244	167	167	177	48	167	192
25	124	9	44	81	125	41	65
71	204	217	180	242	114	30	129
139	36	238	8	9	165	127	178

(b)

Espaços de cor

- Modelos de representação de cores.
- RGB e YCbCr.
- Ambos compostos de três canais.

RGB

- Canais vermelho(R), verde(G) e azul(B).
- Matrizes contém fatores de influência de cada cor na imagem final.

$$C_{i,j} = p_{i,j,1}.R + p_{i,j,2}.G + p_{i,j,3}.B, \quad (1 < i < M, 1 < j < N)$$

RGB



Uma imagem RGB e seus três canais separados.

YCbCr

- Usado em vídeos por sua capacidade de compressão.
- Canais de luminância(Y), crominância azul(Cb) e crominância vermelha(Cr).
- Pode ser obtida através da imagem RGB.
- Canais de crominância são gerados pela diferença entre Y e canais R e B.

$$Y = 0,299.R + 0,587.G + 0,114.B \quad (2)$$

YCbCr



A imagem YCbCr obtida de uma imagem RGB e seus canais.

Descritores de textura

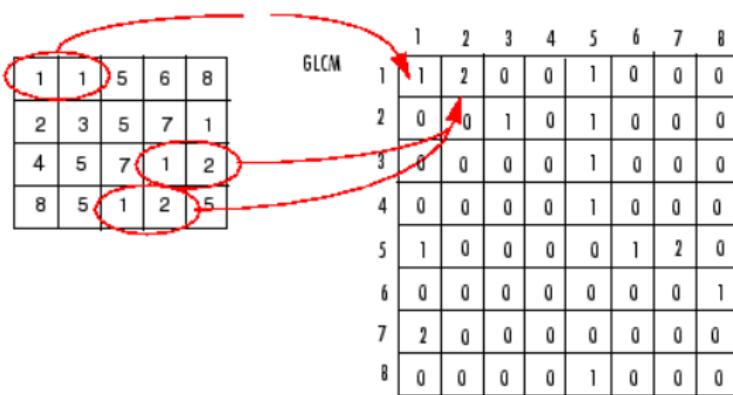
Algoritmos que retorna valores que representam padrões de uma imagem analisada. Analisam a imagem como um todo, ao invés de *pixel-a-pixel*.

GLCM

- *Gray Level Co-occurrence Matrix.*
- Aplicada em imagens de nível de cinza.
- Medidas estatísticas.
- Analisa uma certa relação espacial entre dois *pixels*.
- Para este trabalho, a relação é a vizinhança direita.

GLCM

- Saída do algoritmo é uma matriz $M \times M$ onde M é o maior nível de cinza possível.



Exemplo da elaboração da GLCM. Extraída de
<https://www.mathworks.com/help/images/ref/graycomatrix.html>

GLCM

Medidas

Contraste: $C = \sum_{i,j} (i - j)^2 P_{(i,j)}$

Correlação: $Co = \sum_{i,j} \frac{(i - \mu_i)(j - \mu_j)P_{(i,j)}}{\sigma_i \sigma_j}$

Energia: $E = \sum_{i,j} P_{(i,j)}^2$

Homogeneidade: $H = \sum_{i,j} \frac{P_{(i,j)}}{1 + |i - j|}$

Vídeos

Vídeo

Representação de cenas dinâmicas do mundo real. A sensação de movimento é criada através da exibição de imagens digitais a uma taxa adequada.

Detecção de Movimento

Análise da diferença entre dois quadros para estimar a movimentação de objetos na cena.

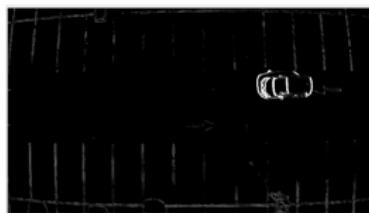
Fluxo óptico

- Estimativa da velocidade do movimento de cada *pixel* da imagem.
- $I(x, y, t) = I(x + dx, y + dy, t + dt)$
- Restrição do Fluxo óptico: $\nabla I v + I_t = 0$.
- É preciso analisar a vizinhança para encontrar ambos os componentes de v .
- Método Lucas-Kanade: vizinhança local.

Fluxo óptico



(a)



(b)

- (a) Representação dos vetores de velocidade estimado pelo fluxo óptico.
(b) Imagem em níveis de cinza onde a intensidade do *pixel* representa a magnitude do seu vetor de velocidade.

Roteiro

- 1 Problema e motivação
- 2 Objetivos
- 3 Fundamentação Teórica
- 4 Redes Neurais Artificiais**
- 5 Solução proposta
- 6 Resultados Experimentais
- 7 Conclusao

Redes Neurais Artificiais

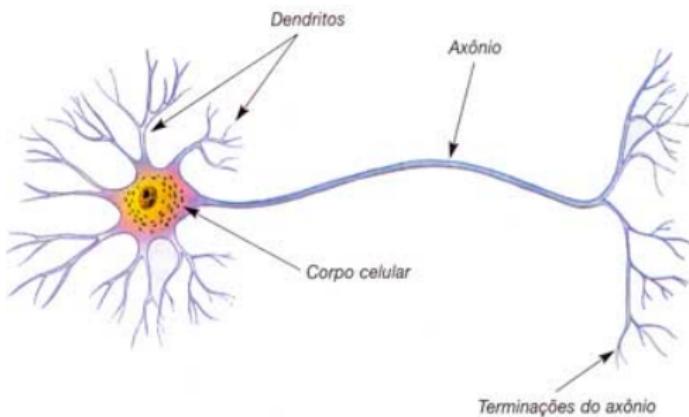
Redes Neurais

Sistemas inspirados no cérebro humano para realização de tarefas como classificação de padrões e ajuste de funções. Usam elementos de processamento distintos que trabalham paralelamente(neurônios).

Haykin define como:

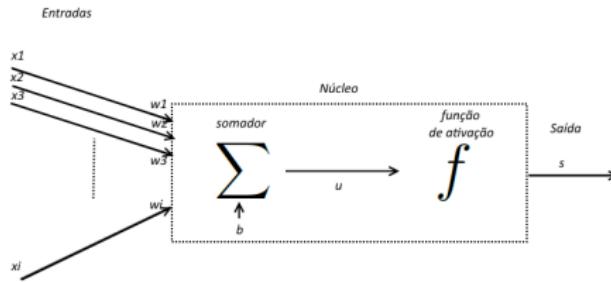
"...um processador distribuído massivamente paralelo composto por unidades simples de processamento, que possui uma propensidade natural a armazenar conhecimento experimental e torná-lo disponível para uso."

Neurônio



A estrutura básica de um neurônio humano

Neurônio Artificial

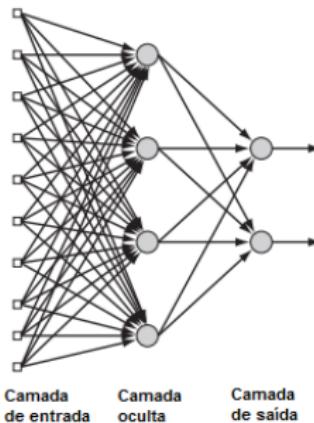


A estrutura básica de um neurônio artificial

Arquitetura *Feed-forward*

- Arquitetura para organização dos neurônios de uma rede neural artificial.
- Camada de entrada, camadas ocultas e camada de saída.
- Informação flui apenas no sentido entrada-saída.
- Cada neurônio de uma camada é ligado a todos da camada seguinte.
- Não há ligação entre neurônios de uma mesma camada.

Arquitetura *Feed-forward*



A arquitetura feed-forward. Adaptada de [?]

Treinamento

- Processo onde a rede é configurada para realizar a tarefa específica para qual foi criada.
- Aprendizado: capacidade da rede de aproximar o comportamento das entradas durante o treinamento.
- Generalização: capacidade da rede de prever e operar sobre entradas que não estavam no conjunto de treinamento.
- Valores dos pesos w e dos deslocamentos b são definidos.

Treinamento

- **Conjunto de treinamento:** Conjunto utilizado para a calibração dos valores de w e b .
- **Conjunto de validação:** Após cada iteração do treinamento, a rede valida os valores configurados usando este conjunto.
- **Conjunto de testes:** A rede é apresentada a este conjunto ao final do treinamento para verificação do funcionamento.

Treinamento

Treinamento Supervisionado

Elementos dos conjuntos de treinamento e validação são as entradas e os gabaritos da saídas desejadas de cada entrada. Treina até que a saída seja suficientemente próxima do gabarito.

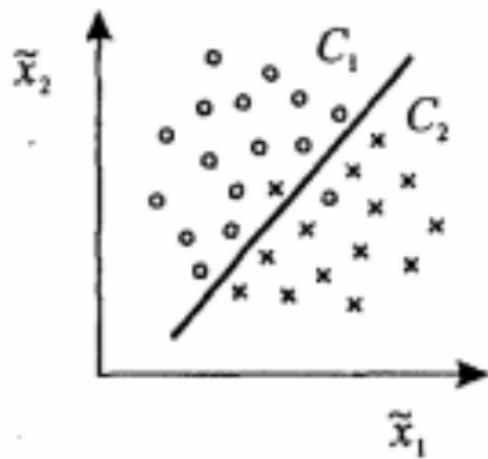
Treinamento Não-Supervisionado

A rede recebe um conjunto de entradas e aprende características intrínsecas dos dados apresentados.

Classificação de padrões

- Consiste em analisar padrões de um conjunto de dados e designar o conjunto a uma classe pré-determinada.
- A rede analisa características ou *features*.
- *Features* devem ser descritivos e discriminantes.
- Limiar de decisão.
- A rede determina a probabilidade de que uma entrada pertença a uma classe.

Classificação de padrões

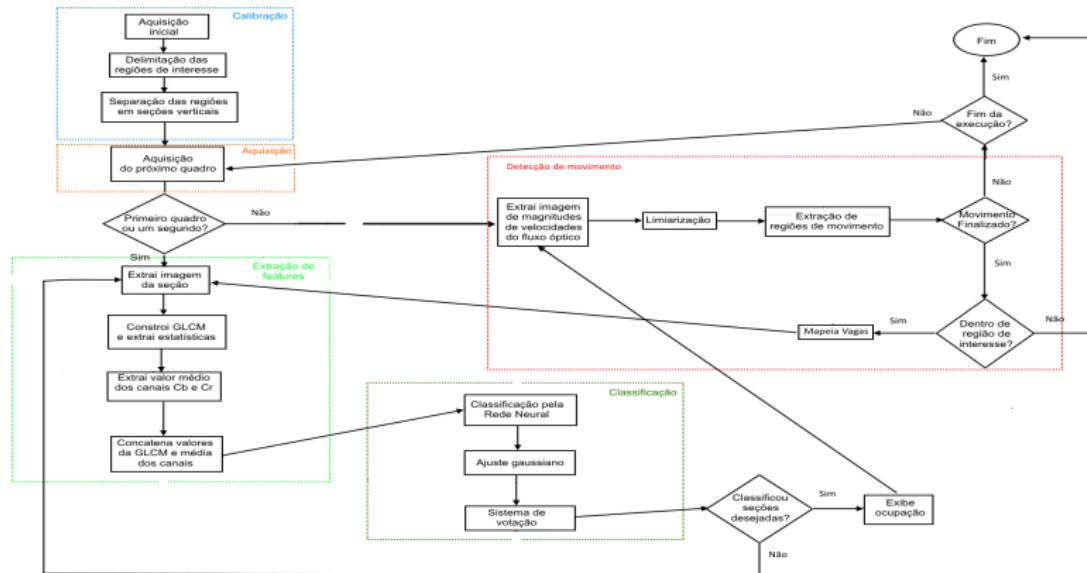


Um gráfico que mostra duas características \tilde{x}_1 e \tilde{x}_2 de um problema de classificação hipotético. Extraída de [?].

Roteiro

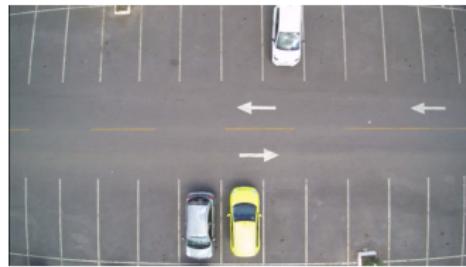
- 1 Problema e motivação
- 2 Objetivos
- 3 Fundamentação Teórica
- 4 Redes Neurais Artificiais
- 5 Solução proposta
- 6 Resultados Experimentais
- 7 Conclusão

Fluxograma



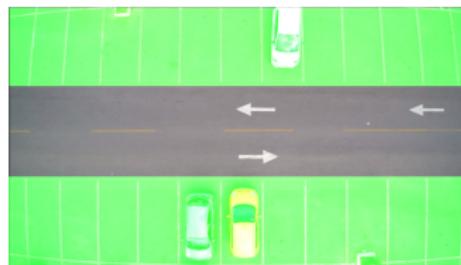
Aquisição

Imagens de câmeras montadas em postes de luz ou similar.



Regiões de interesse

- Definidas no momento de instalação.
- Não devem ter intersecção.
- Regiões onde há vagas na imagem.



Seções verticais

Cada ROI é dividida em um número igual de seções verticais.



Classificação das seções

A classificação tem 4 etapas:

- Extração das características.
- Classificação pela Rede Neural.
- Ajuste pelos vizinhos.
- Sistema de votação

Extração das características

Imagens individuais de cada seção são extraídas.



Extração das características

- Medidas GLCM: $\begin{pmatrix} \text{Contraste} \\ \text{Correlao} \\ \text{Energia} \\ \text{Homogeneidade} \end{pmatrix}$
- Médias dos canais de crominância: $\begin{pmatrix} MCb \\ MCr \end{pmatrix}$
- Vetor final de entrada: $\begin{pmatrix} \text{Contraste} \\ \text{Correlao} \\ \text{Energia} \\ \text{Homogeneidade} \\ MCb \\ MCr \end{pmatrix}$

Classificação da Rede Neural Artificial

- Duas classes possíveis: carro(1) ou vaga (2).
- Rede neural recebe vetor de entrada e retorna vetor com probabilidade de cada classe.

$$\begin{pmatrix} P_1 \\ P_2 \end{pmatrix} \quad (3)$$

Ajuste dos vizinhos

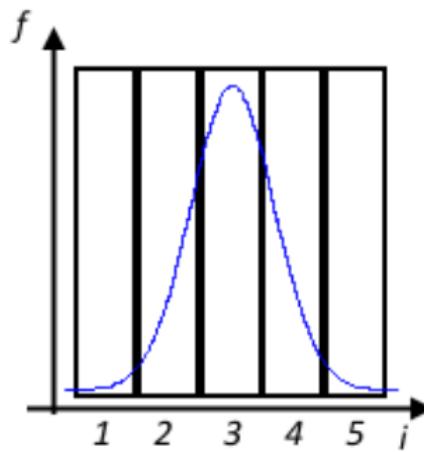
A classe das seções vizinhas influencia a classificação de uma seção de acordo com as equações:

$$f(s, v) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp^{-\frac{(i_s - i_v)^2}{2\sigma^2}} \quad (4)$$

$$P_{cv} = P_{cv}(1 + f(s, v)) \quad (5)$$

$$P_{\bar{cv}} = P_{\bar{cv}}(1 - f(s, v)) \quad (6)$$

Ajuste dos Vizinhos



Ajuste dos Vizinhos

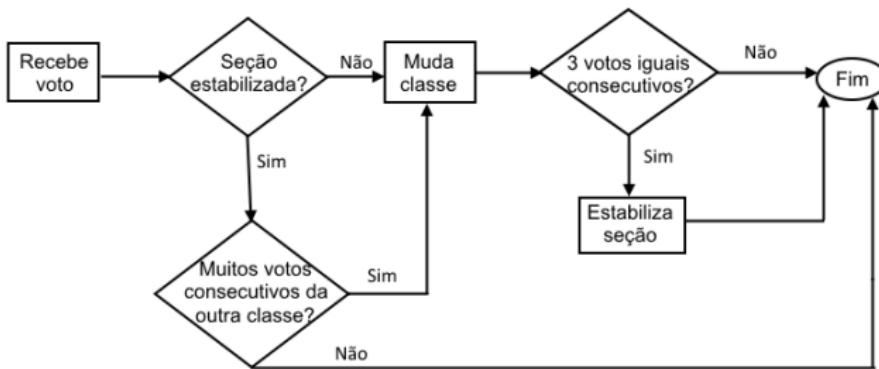
Ajuda a evitar erros como este:



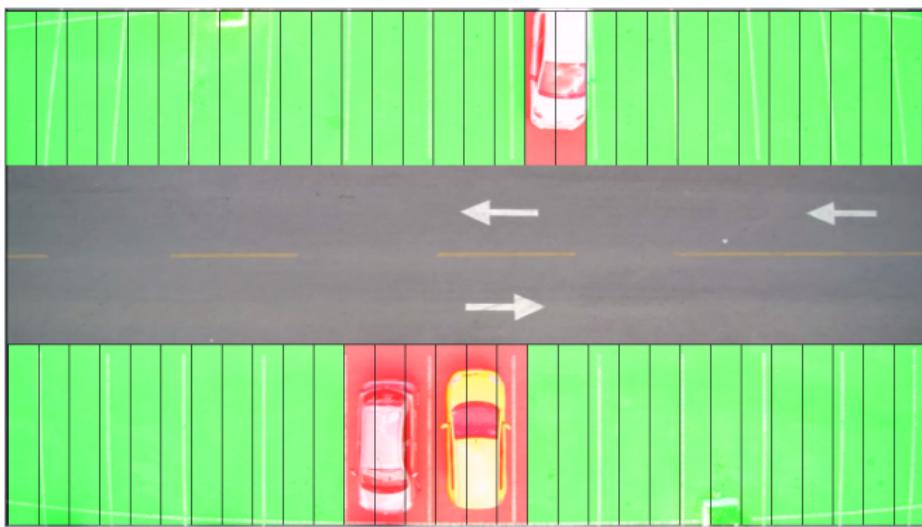
Sistema de votação

- Votação define se a seção está estabilizada ou não.
- Voto é baseado no vetor de probabilidades modificado pelo ajuste dos vizinhos.
- $voto = \begin{cases} 1, & \text{se } P_1 \geq 0,6 \\ 2, & \text{caso contrário} \end{cases}$
- Seção não muda de classe se estiver estabilizada.
- Evitar flutuações na ocupação das seções.

Sistema de votação



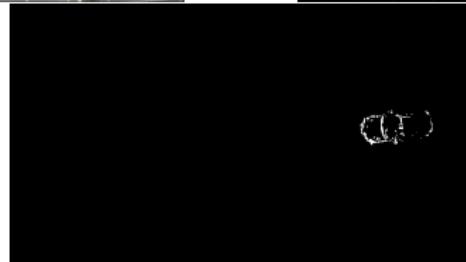
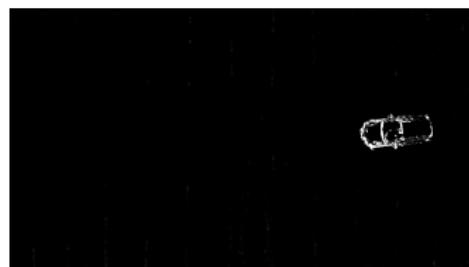
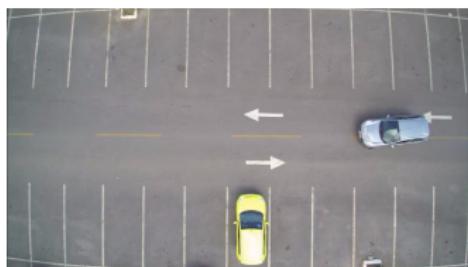
Seções classificadas



Detecção de movimento

- Movimento detectado pelo método do Fluxo Óptico
- Resultado: imagem que representa as magnitudes dos vetores velocidade
- Essa imagem é limiarizada para eliminar os vetores não significativos.

Detecção de movimento



Regiões do movimento

Regiões são detectadas através de análise dos pixels brancos da imagem limiarizada.



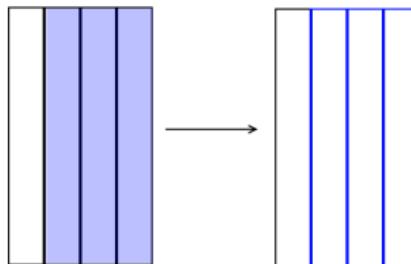
Mapeamento das vagas

- No momento inicial da execução conjuntos de seções ocupadas consecutivas formam uma vaga.
- Quando um movimento termina, a posição aonde ele acabou e começou é analisada.
- As seções verticais interceptadas pelo movimento nestas posições ficam não-estabilizadas, são reclassificadas e marcadas como uma vaga.
- Assume-se que os carros estão parando em vagas corretamente.

Tratamento de intersecção de vagas

Modo simples

Seções marcadas pelo movimento passam a fazer parte de uma vaga nova, vaga antiga é composta das seções restantes.



Tratamento de intersecção de vagas

Modo Complexo

Segue um conjunto mais elaborado de regras:

- $S_m \supseteq V \Rightarrow V = S_m$
- $S_m \subset V \Rightarrow V = S_m$. Caso $S_m - V$ seja um conjunto contínuo de um número suficiente de seções, uma nova vaga V_2 também é criada.
- Se $S_m \cap V \neq \emptyset$ as seções da interseção são divididas entre as duas vagas, dando preferência para S_m .

Roteiro

- 1 Problema e motivação
- 2 Objetivos
- 3 Fundamentação Teórica
- 4 Redes Neurais Artificiais
- 5 Solução proposta
- 6 Resultados Experimentais**
- 7 Conclusão

Aquisição dos vídeos

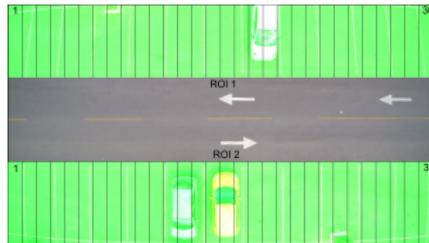
Drone Yuneec Typhoon Q500+.



ring

Experimentos

- Oito vídeos curtos com ROIs determinadas previamente divididas em 30 seções verticais.
- Desempenho do programa foi comparado com resultados de três observadores humanos: F, M e P.



Instruções

Cada observador recebeu as seguintes instruções:

- As regiões de interesse e as seções são numeradas como indicado na figura.
- Uma seção ocupada é aquela cuja maior parte de sua área está ocupada por um veículo.
- No momento inicial do vídeo (0 segundos), indique quais seções verticais estão ocupadas através do número das ROI e das seções, as outras serão assumidas como livres. Indique também o número de vagas ocupadas.
- Se a qualquer momento o estado de ocupação de uma seção mudar, indique o tempo da mudança, a seção onde ocorreu a mudança e a natureza da mudança(ocupada ou liberada).

Experimentos

Ocupação das seções

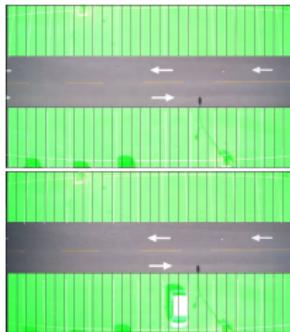
- Um acerto ocorre se em um determinado momento t , o programa e o observador concordam quanto a ocupação da seção.
- 60 acertos possíveis por segundo.
- Taxa de acerto = número total de acertos/acertos possíveis.

Vagas ocupadas

- Um acerto ocorre se em um determinado momento t , o programa e o observador concordam quanto ao número de vagas ocupadas.
- Número de acertos possíveis é igual a duração do vídeo.

Vídeo 1

No primeiro caso de testes, o vídeo começa com um estacionamento vazio. Depois de alguns segundos um único veículo de cor branca entra na cena pela direita e estaciona em uma vaga na parte inferior da imagem. O vídeo tem 15s de duração.



Vídeo 1

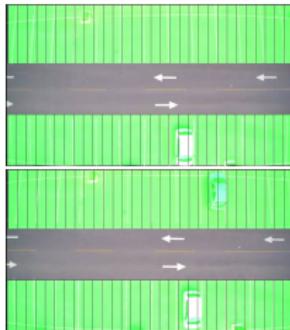
Acertos - seções		
Observador	Acertos	Taxa de acertos
F	886	98,44%
P	886	98,44%
M	884	98,22%
Média	885,3	98,37%

Vídeo 1

Acertos - vagas		
Observador	Acertos	Taxa de acertos
F	1	7%
P	1	7%
M	1	7%
Média	1	7%

Vídeo 2

Neste vídeo, um carro branco está estacionado no conjunto inferior de vagas. Um veículo cinza entra pela direita e estaciona no conjunto superior de vagas. O vídeo tem uma duração de 10s.



Vídeo 2

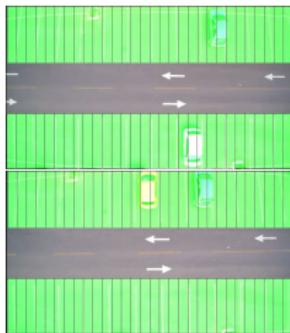
Acertos - seções		
Observador	Acertos	Taxa de acertos
F	586	97,66%
P	586	97,66%
M	586	97,66%
Média	586	97,66%

Vídeo 2

Acertos - vagas		
Observador	Acertos	Taxa de acertos
F	10	100%
P	10	100%
M	10	100%
Média	10	100%

Vídeo 3

Neste vídeo, dois carros se encontram no estacionamento. Um veículo amarelo entra pela direita e estaciona na seção superior das vagas. O veículo branco sai pela parte de baixo da tela. Ele tem duração de 20s.



Vídeo 3

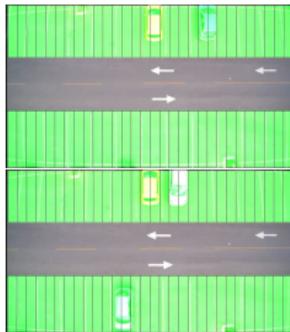
Observador	Acertos	Taxa de acertos
F	1165	97,08%
P	1113	92,75%
M	1136	94,66%
Média	1138	94,83%

Vídeo 3

Acertos - vagas		
Observador	Acertos	Taxa de acertos
F	19	95%
P	11	55%
M	18	90%
Média	16	80%

Vídeo 4

Neste vídeo, o veículo branco entra em cena pela parte superior da tela e estaciona entre os outros dois carros. O carro cinza sai da vaga que ocupava e estaciona em uma na área inferior. Duração de 30s.



Vídeo 4

Acertos - seções		
Observador	Acertos	Taxa de acertos
F	1750	97,22%
P	1749	97,16%
M	1772	98,44%
Média	1757	97,61%

Vídeo 4

Acertos - vagas		
Observador	Acertos	Taxa de acertos
F	22	73,33%
P	22	73,33%
M	24	80%
Média	22,66	75,55%

Vídeo 5

Um veículo desocupa a sua vaga saindo pela parte superior da tela.
Duração de 10s.



Vídeo 5

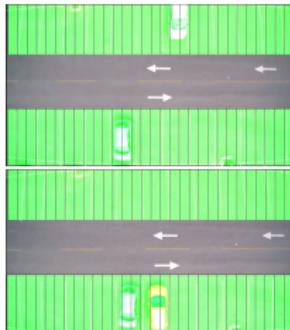
Acertos - seções		
Observador	Acertos	Taxa de acertos
F	597	99,50%
P	585	97,50%
M	587	97,83%
Média	1757	97,61%

Vídeo 5

Acertos - vagas		
Observador	Acertos	Taxa de acertos
F	10	100%
P	9	90%
M	10	100%
Média	9,66	9%

Vídeo 6

Neste vídeo o veículo amarelo entra na cena pela parte inferior da tela. O veículo branco desocupa sua vaga e sai da tela pelo lado esquerdo. O vídeo tem 13s.



Vídeo 6

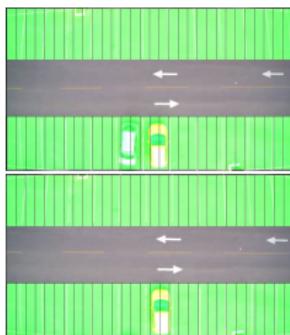
Acertos - seções		
Observador	Acertos	Taxa de acertos
F	752	96,40%
P	774	99,23%
M	767	98,33%
Média	764,33	97,99%

Vídeo 6

Acertos - vagas		
Observador	Acertos	Taxa de acertos
F	10	76,92%
P	11	84,61%
M	9	69,23%
Média	10	76,92%

Vídeo 7

Neste vídeo um dos carros sai da cena pela esquerda. O vídeo tem 17s.



Vídeo 7

Acertos - seções		
Observador	Acertos	Taxa de acertos
F	1013	99,31%
P	1014	99,41%
M	1016	99,60%
Média	1014,33	99,44%

Vídeo 7

Acertos - vagas		
Observador	Acertos	Taxa de acertos
F	16	94,11%
P	16	94,11%
M	17	100%
Média	16,33	96,07%

Vídeo 7

No oitavo e último caso de testes, um carro branco passa pela região central da imagem sem estacionar em nenhuma vaga e depois um carro cinza estaciona na região inferior. O vídeo tem 23s de duração.



Vídeo 8

Acertos - seções		
Observador	Acertos	Taxa de acertos
F	1376	99,71%
P	1379	99,92%
M	1379	99,92%
Média	1378	99,85%

Vídeo 8

Acertos - vagas		
Observador	Acertos	Taxa de acertos
F	23	100%
P	23	100%
M	23	100%
Média	23	100%

Roteiro

- 1 Problema e motivação
- 2 Objetivos
- 3 Fundamentação Teórica
- 4 Redes Neurais Artificiais
- 5 Solução proposta
- 6 Resultados Experimentais
- 7 Conclusao

Conclusao

Perguntas?

Referencias |

- [1] DBL Bong, KC Ting, and KC Lai. Integrated approach in the design of car park occupancy information system (coins). *IAENG International Journal of Computer Science*, 35(1):7–14, 2008.
- [2] Wen-Yuan Chen and Chuin-Mu Wang. The dynamic background generation scheme using an image frame statistical comparison method. *International Journal of Advancements in Computing Technology*, 4(18), 2012.
- [3] José Eustáquio Rangel de Queiroz and Herman Martins Gomes. Introdução ao processamento digital de imagens. *RITA*, 13(2):11–42, 2006.

Referencias II

- [4] Diana Delibaltov, Wencheng Wu, Robert P Loce, Edgar Bernal, et al. Parking lot occupancy determination from lamp-post camera images. In *Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on*, pages 2387–2392. IEEE, 2013.
- [5] Rafael C Gonzalez. *Digital image processing*. Pearson Education India, 2009.
- [6] Ana Beatriz Vicentim Graciano. *Rastreamento de objetos baseado em reconhecimento estrutural de padroes*. PhD thesis, Universidade de São Paulo, 2007.

Referencias III

- [7] Zhai Hai-tao, Wu Jian, Xia Jie, and Cui Zhi-ming. Self-adaptive detection of moving vehicles in traffic video. In *The 2009 International Symposium on Web Information Systems and Applications (WISA 2009)*, page 449, 2009.
- [8] IBGE. *Introdução ao processamento digital de imagens*. IBGE, 2000.
- [9] Ogê Marques Filho and Hugo Vieira Neto. *Processamento digital de imagens*. Brasport, 1999.
- [10] Idris M.Y.I, Leng Y.Y, et al. Car park system: A review of smart parking system and its technology. *Information Technology Journal*, 8(8):101–113, June 2009.

Referencias IV

- [11] Nicholas True. Vacant parking space detection in static images. *University of California, San Diego*, 2007.
- [12] AI VKL. Jain, "fundamentals of digital image processing," , 1989.

Conclusão

[1] [2] [3] [4][5] [6] [7] [8] [10] [9] [11]
[12]