



Detecção de veículos

UMA ABORDAGEM POR FUSÃO

On-road Vehicle Detection Fusing Radar and Vision

Autores: Xin Liu, Zhenping Sun, Hangen He

The Institute of Automation National University of Defense Technology

Changsha, China

Detecção de veículos: por quê?

- ▶ Sistemas de segurança veicular
 - ▶ Veículos com detecção de colisão frontal;
 - ▶ Controle de velocidade de cruzeiro adaptativo.
- ▶ Sistemas de controle de tráfego
 - ▶ Detecção de fluxo;
 - ▶ Possível controle de fluxo (atuadores);

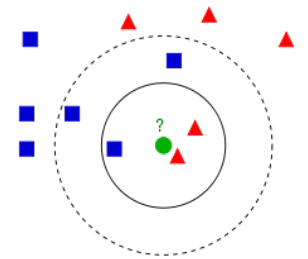
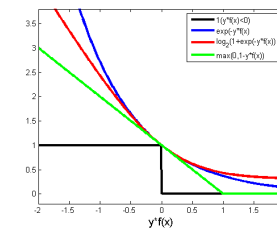
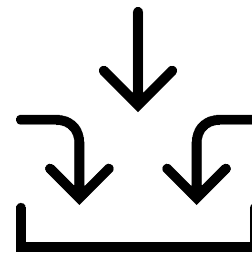
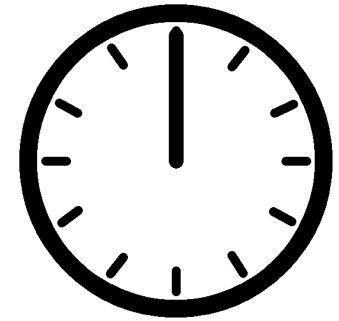
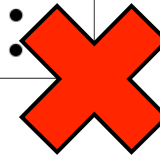
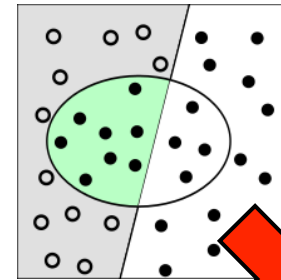
Detecção de veículos: como?

Problemas

- ▶ Sistema deve ser robusto/confiável
- ▶ Tempo real

Soluções

- ▶ Fusão: dois sensores
- ▶ Abordagens rápidas
 - ▶ SVM polinomial de segundo grau
 - ▶ KNN



Referencial teórico

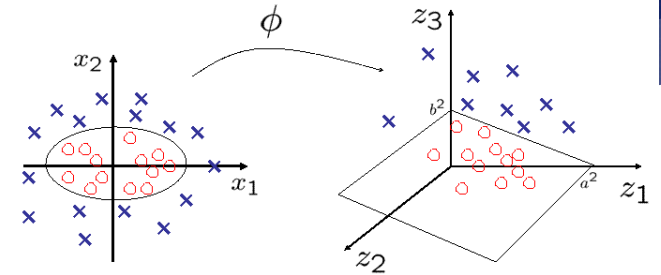
TÉCNICAS UTILIZADAS

Vehicle Shadow Segmentation (VSS)

- ▶ Investigando-se a intensidade da imagem, percebeu-se que a sombra sob o veículo é mais escura do que outras áreas da via pavimentada;
- ▶ A partir da segmentação da sombra e subsequente detecção de bordas, é possível predizer se aquela sombra é ou não de um veículo com tais informações;
- ▶ O autor utiliza essa já fomentada técnica de detecção de veículos, pois é rápida e já foi amplamente testada, porém propõe um macroalgoritmo específico.

<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.45.3234>

Support Vector Machines (SVM)



- ▶ Modelo supervisionado de Aprendizagem de Máquina usado para classificação de dados;
- ▶ Basicamente, se utiliza de um mecanismo matemático (kernel trick) que, através de uma função (i.e. kernel function) mapeia os inputs de dados em um espaço com mais dimensões;
- ▶ O autor faz uso dessa técnica pela robustez apresentada, porém a função kernel proposta é quadrática a fim de reduzir o custo computacional da classificação.

https://en.wikipedia.org/wiki/Support_vector_machine

Nearest Neighbor Clustering (NNC)

- ▶ Algoritmo *clássico* para clusterização que leva em consideração apenas qual próximos os elementos estão;
- ▶ Custo computacional extremamente baixo;

https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

Metodologia

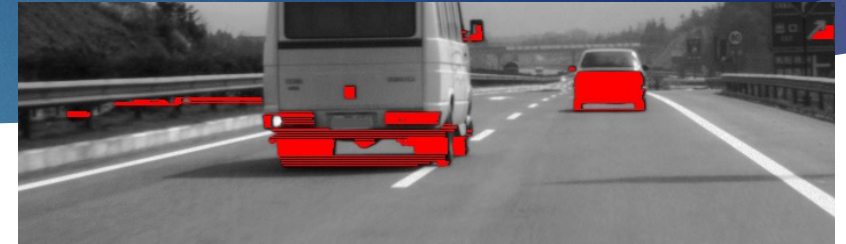
PROPOSTA DO AUTOR PARA
RESOLUÇÃO DO PROBLEMA

Detecção da sombra de veículo

- ▶ Característica importante: há uma restrição quanto a largura;
- ▶ Característica: VSS
 - ▶ Definição: $vss(u_{\downarrow L}, u_{\downarrow R}, v)$
 - ▶ $u_{\downarrow L}$: ponto terminal esquerdo da linha (abcissa)
 - ▶ $u_{\downarrow R}$: ponto terminal direito da linha (abcissa)
 - ▶ v : altura da linha no plano (ordenada)
 - ▶ Condições:
 - ▶ $\sum_{u=u_{\downarrow L} \uparrow u_{\downarrow R}} P(u, v) / (u_{\downarrow L} - u_{\downarrow R}) < P_{\downarrow T}$
 - ▶ $HSobel(u_{\downarrow L}, v) < -S_{\downarrow T}; HSobel(u_{\downarrow R}, v) > S_{\downarrow T}$
 - ▶ $L_{\downarrow L}(v) \leq (u_{\downarrow L} - u_{\downarrow R}) \leq L_{\downarrow H}(v)$



Geração de hipótese de veículo

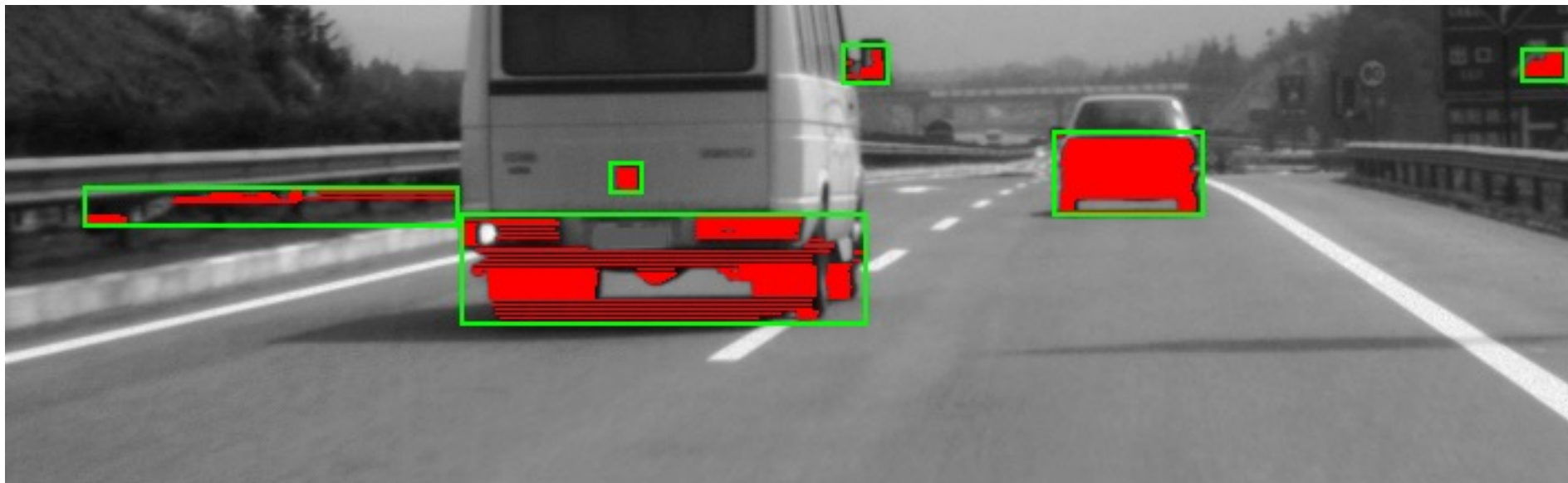
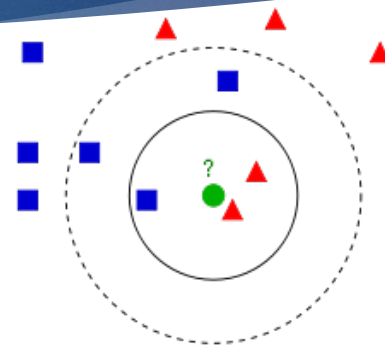


► Extração de VSS

1. Varre pixels da imagem na linha v ,
 1. Se um pixel satisfaz $P(u, v) < P_{\downarrow T}$ e $HSobel(u, v) < -S_{\downarrow T}$, coloca-o na set NE
 2. Se um pixel satisfaz $P(u, v) < P_{\downarrow T}$ e $HSobel(u, v) > S_{\downarrow T}$, coloca-o na set PE
2. Para cada pixel $X(u_{\downarrow L}, v)$ da set NE, checa os pixels na set PE um por um;
 1. Se o pixel $Y(u_{\downarrow R}, v)$ satisfaz $\sum_{u=u_{\downarrow L}}^{u_{\downarrow R}} P(u, v) / (u_{\downarrow L} - u_{\downarrow R}) < P_{\downarrow T}$ e $L_{\downarrow L}(v) \leq (u_{\downarrow L} - u_{\downarrow R}) \leq L_{\downarrow H}(v)$, $vss(u_{\downarrow L}, u_{\downarrow R}, v)$ existe, próximo passo, ignorando outros pixels em PE;
3. Se existe $vss(u_{\downarrow L}, u_{\downarrow R}, v)$ no passo anterior, cancela todos os pixels que satisfazem $u \leq u_{\downarrow y}$ na set NE;
4. $v =$ próxima linha da imagem (itera por todas as linhas)

Geração da hipótese do veículo

- ▶ Agrupamento das *features*
 - ▶ Número de grupos é desconhecido para cada imagem;
 - ▶ Algoritmo kNN (desempenho)
 - ▶ Geração das bounding boxes



Geração da hipótese do veículo

- ▶ Validação das bounding boxes
 1. A largura destas deve satisfazer a restrição de largura própria de veículos;
 1. $L \downarrow L(v) \leq (u \downarrow L - u \downarrow R) \leq L \downarrow H(v)$
 2. A proporção de largura/altura deve ser menor que 10:1;



Fusão

- ▶ Matching (SVM)

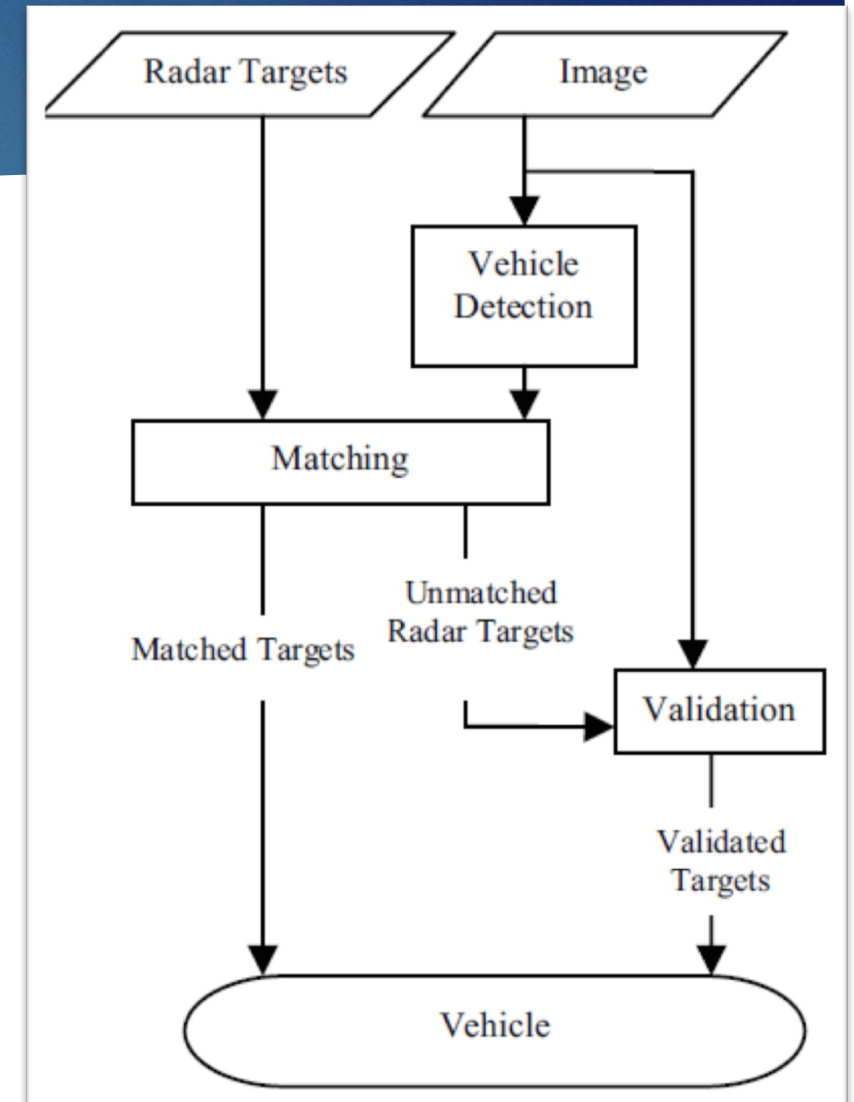
- ▶ Transformar a informação do radar para o sistema de referência da visão;

$$C(p_R, p_C) = \alpha |u_R - u_C| + \beta |v_R - v_C|$$

- ▶ Classificar

- ▶ Validação

- ▶ Elementos advindos do radar não validados após fusão são [re]validados.





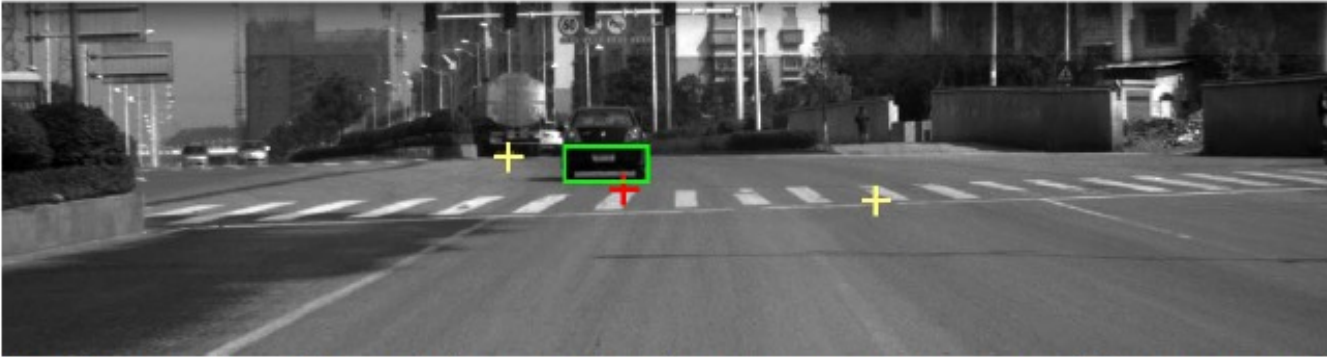
Resultados Experimentais

RESULTADOS APÓS EXECUÇÃO
DA TÉCNICA PROPOSTA EM
DATASETS REAIS

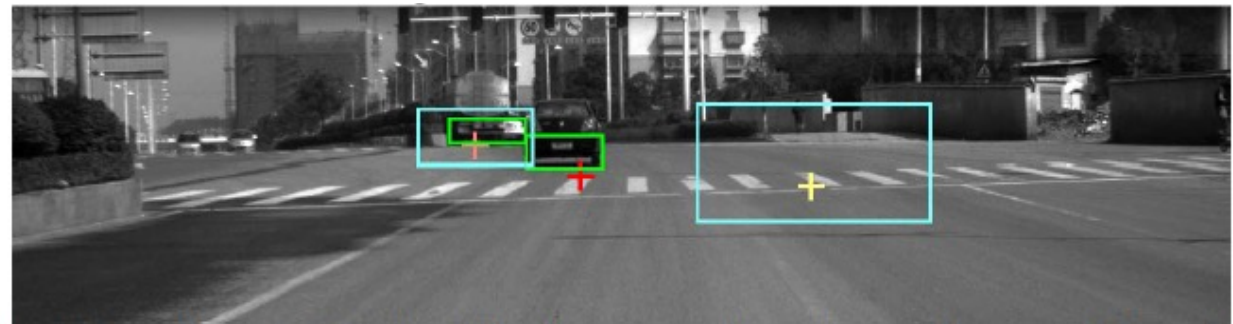
Dados experimentais

- ▶ O dataset de treino contem 800 exemplos positivos e 1500 exemplos negativos;
- ▶ O dataset de teste foi obtido dirigindo por rodovias e vias urbanas, composto de mais de 5000 imagens e a correspondente informação do radar;
- ▶ Os sensores de teste foram
 - ▶ Radar: 120m de alcance, 76Ghz de frequência, detecta até 64 alvos;
 - ▶ Câmera: resolução 1024x384, detecção da visão de até 120m;
- ▶ Os testes foram executados numa máquina baseada em AMD Athlon64 3000+;

Resultados do experimento



(a) The results of matching. Red radar target and green vision target are matched. Yellow radar targets are unmatched.



(b) The results of validation. Blue boxes are the regions of interest. Pink radar target is validated and Yellow radar target is not.

Resultados do experimento

	Road	Curve	Traffic	Other
Scene 1	freeway	Yes	normal	/
Scene 2	freeway	no	Normal	uphill
Scene 3	freeway	no	Normal	bridge shadow
Scene 4	urban	no	Heavy	extremely heavy traffic

Taxa de atualização: 15Hz

	Frames	Vehicles	Radar Detected Targets	Radar False Positive	Matched Targets	Validated Targets	Matched False Positive	Validated False Positive
Scene 1	1766	3045	3021	451	2643	328	3	12
Scene 2	1245	2587	2561	307	2017	490	2	13
Scene 3	1305	2543	2535	346	1993	493	4	22
Scene 4	786	3678	3579	323	2717	752	15	61
Total	5102	11853	11696 (98.7%)	1427 (10.9%)	11433 (96.5%)		132 (1.1%)	

Conclusão

Trata-se de um mecanismo importante para o controle de tráfego (aplicação na engenharia de tráfego) e para a evolução de novas tecnologias na indústria de veículos (e.g. direção autônoma, prevenção de acidentes). Observa-se que a taxa de erros deve ser minimizada, e tal feito, em muitos dos casos é conseguido usando fusão de *inputs*.