

# ANÁLISE DE ATRIBUTOS DE IMAGEM EM SEGMENTAÇÃO BASEADA EM BLOCOS UTILIZANDO REDES NEURAIIS ARTIFICIAIS

Gustavo Buzogany Eboli, Patrick Yuri Shinzato, Denis Fernando Wolf

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

gbuzogany@grad.icmc.usp.br, {shinzato,denis}@icmc.usp.br

**Resumo** – Navegação é um grande tema que vem recebendo atenção considerável da comunidade científica na área de robótica móvel. A fim de executar uma direção autônoma em ambientes externos, como ruas e estradas, é necessário que o veículo identifique partes do terreno que podem ser percorridas e partes que devem ser evitadas. Este artigo descreve uma análise de identificação de terreno baseada em diferentes informações utilizando redes neurais artificiais MLP. Os testes experimentais usando um carro equipado com uma câmera de vídeo foram realizados em cenários reais para avaliar os atributos propostos e RNAs.

**Palavras-chave** – Navegação, redes neurais artificiais, visão computacional, identificação do caminho, detecção do horizonte.

**Abstract** – Navigation is a broad topic that has been receiving considerable attention from the mobile robotic community. In order to execute a autonomous driving on outdoors, like street and roads, it is necessary that the vehicle identify parts of the terrain that can be traversed and parts that should be avoided. This paper describes an analyses of an terrain identification based on different visual information using a multi-layer perceptron neural network. Experimental tests using a car and a video camera have been conducted in real scenarios to evaluate the proposed features and networks.

**Keywords** – Navigation, artificial neural networks, computer vision, path recognition, horizon detection.

## 1. INTRODUÇÃO

A navegação autônoma é um dos principais problemas da robótica móvel, sendo um requisito para que outras tarefas possam ser executadas. A navegação em ambientes externos como vias públicas e terrenos desconhecidos consiste em um problema muito mais complexo que em ambientes internos. Além do desvio de obstáculos, é necessário que o robô identifique a região onde pode navegar. Graças ao avanço da tecnologia, mais especificamente o aumento de memória e poder de processamento de computadores, pode-se utilizar câmeras de vídeo como sensor para a identificação da região navegável.

Algoritmos de *Road Following* são técnicas desenvolvidas para localizar a rua ou superfície onde o robô poderá navegar. Esses algoritmos normalmente utilizam dados obtidos de sensores de vídeo, ou seja, dados de uma sequência de imagens. Algoritmos de *Road Detection* também têm a função de localizar a superfície navegável, entretanto, nesta classe de algoritmos, a via navegável pode estar em qualquer região da imagem enquanto em algoritmos de *Road Following* assume-se que o robô está na área navegável quando o sistema é iniciado [1].

Em [2], as técnicas de *Road Following* são divididas em duas classes: uma onde os métodos utilizam todos os pixels abaixo da linha do horizonte e outra onde os métodos utilizam os pixels dentro de uma região predita - normalmente essa região é uma estimativa da área ocupada pela rua na imagem. Em nosso trabalho, também utilizamos o método de classificação baseado em blocos para classificar a região da imagem que é composta por céu, sendo assim possível estimar a posição da linha do horizonte. Em [1], as técnicas são divididas em 3 classes de abordagens: técnicas de *Road Following* que procuram pelas faixas da rua, [3], [4] e [5]; técnicas de detecção de bordas usando gradiente, [6] e [7]; e por fim sistemas capazes de tirar proveito da aparência homogênea de ruas, como SCARF [8].

Outras abordagens na área de processamento de imagens vêm obtendo bons resultados em algoritmos de *Road Following*, como [9], [10], [11], [12] e [13]. Um dos mais recentes é o de [1] que apresentou um modelo baseado em cor com aprendizado em tempo real para determinação de rua, onde seu método busca aprender a aparência homogênea da estrada.

Um dos maiores problemas em se trabalhar com imagens para sistemas de resposta em tempo real, é a quantidade de dados - normalmente milhares de pixels - para serem processados. Este trabalho apresenta uma análise da eficiência da classificação de regiões navegáveis e não navegáveis em uma imagem, utilizando combinações de atributos os quais foram gerados a partir de uma segmentação baseada em blocos. A metodologia de classificação foi baseada em redes neurais artificiais do tipo *multi-layer perceptron* (MLP).

## 2. MÉTODO DE CLASSIFICAÇÃO BASEADO EM BLOCOS

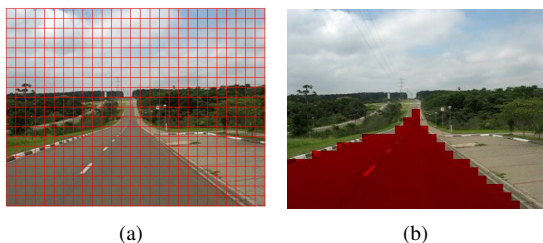
Este método consiste em tratar e avaliar uma porção de pixels diretamente conectados (adjacentes) como um grupo, ou seja, blocos de pixels. A partir deste bloco é gerado um único valor ou característica para representá-lo. Esta característica pode ser



Figura 1: Veículo utilizado na coleta de dados.

a média dos valores RGB, entropia dos pixels ou muitas outras medidas. Este método propõe fatiar a imagem em quadrados de dimensão  $(K \times K)$ , como mostra a Figura 2(a), onde cada quadrado é um bloco.

Suponha a matriz  $I$  uma imagem de resolução igual a  $(M \times N)$ . O valor do elemento  $I(m, n)$  corresponde ao valor do pixel na  $m$ -ésima linha e  $n$ -ésima coluna da imagem, onde  $(0 \leq m < M)$  e  $(0 \leq n < N)$ . Assim, o bloco  $B(i, j)$  contém todos os pixels  $I(m, n)$  tal que  $((i * K) \leq m < ((i * K) + K))$  e  $((j * K) \leq n < ((j * K) + K))$ .



(a)

(b)

Figura 2: Segmentação baseada em blocos. (a) Imagem fatiada em pequenos quadrados. (b) Exemplo de uma imagem classificada onde a região considerada navegável está pintada de vermelho.

Assim, se o valor da característica que representa o bloco  $B(i, j)$  for considerado como um valor que representa uma região navegável na imagem então todos os pixels pertencentes a  $B(i, j)$  serão considerados navegáveis, a Figura 2(b) mostra os pixels dos grupos considerados navegáveis em vermelho.

## 2.1. MEDIDAS ESTATÍSTICAS UTILIZADAS COMO ATRIBUTOS DE BLOCOS

Diversas medidas estatísticas foram utilizadas para representar os pixels de um bloco. Medidas simples, como média e probabilidade, e medidas mais complexas, como entropia e variância, foram geradas. Suas definições são descritas a seguir:

1. **Entropia:** Famosa na área de teoria de informação a entropia de Shannon pode ser definida, de maneira simplificada, como grau de regularidade de um conjunto de dados [14]. Matematicamente a entropia de Shannon é definida como:

$$E(X) = - \sum_{x \in X} p(x) \log_2(p(x))$$

Onde  $p(x)$  é a probabilidade de  $x$  aparecer no conjunto  $X$ . Neste trabalho,  $x$  é um pixel e  $X$  é um bloco.

2. **Energia:** Essa medida representa uma relação entre valores altos em um grupo. Pode ser descrita como:

$$\varepsilon = \sum_{i=0}^{C-1} (p(x))^2 \quad (1)$$

3. **Variância:** Esse valor é muito conhecido na estatística, representa a dispersão em relação a média. Pode ser descrita como:

$$\sigma^2 = \sum_{i=0}^{C-1} (x - \mu)^2 * p(x) \quad (2)$$

Onde  $C$  é o número de cores.

### 3. DETERMINAÇÃO DA POSIÇÃO DA LINHA DO HORIZONTE

Para otimizar os resultados da identificação da região navegável, utilizamos o método de classificação baseado em blocos para determinar a posição da linha do horizonte. Assim, toda a região da imagem acima da linha do horizonte não é processada pelo classificador da região navegável, reduzindo o custo computacional do processamento das imagens e fazendo com que a saída da região navegável não tenha nenhum ruído acima desta linha.

Aplicando cada bloco  $B(i, j)$  da imagem na RNA de classificação do céu, é gerada uma matriz  $S_{nm}$  com as saídas da rede. Se o valor da saída for superior a um valor de *threshold*, então essa posição  $(i, j)$  é definida como céu. Para determinar a posição da linha do horizonte, é feito o cálculo da proporção  $p$  da imagem que é coberta por céu. Assim, a posição relativa da linha do horizonte é igual à altura  $h$  da imagem, multiplicada por  $p$ . Como a posição da linha do horizonte não se altera significativamente em intervalos pequenos de tempo, a determinação da posição da linha do horizonte só ocorre a cada 20 frames.

### 4. EXPERIMENTOS E RESULTADOS

A fim de analisar várias combinações de atributos, foram coletados dados em ambientes reais sob diferentes condições. Mais especificamente, foi filmado um caminho percorrido por um carro. Esse caminho era composto por ruas ladeadas de calçadas, estacionamento e vegetação, além disso, porções da rua, em alguns trechos, apresentaram condições adversas como sujeira e rastros de outros carros.

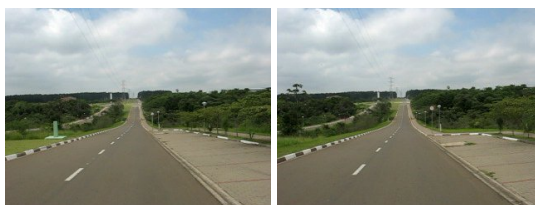
No experimento, utilizamos um carro equipado com uma câmera digital Canon A610. A resolução da imagem era de (320 x 240) pixels a 30 FPS. O carro e a câmera foram utilizados apenas para a coleta de dados. Para treinar e executar todas as RNAs foi utilizado o software *Stuttgart Neural Network Simulator* (SNNS). A biblioteca OpenCV [15] foi utilizada no programa em linguagem de programação C, tanto para importação da imagem como para visualização dos resultados das RNAs. O tamanho do bloco utilizado foi de  $K = 10$ , o que significa que cada frame possui 768 blocos.

Neste trabalho foram gerados 21 atributos: média de R (vermelho), média de G (verde), média de B (azul), média de H (*hue*), média de S (saturação), média de V (brilho), entropia de HSV, entropia de H, entropia de S, entropia de V, variância de H, variância de S, variância de V, energia de RGB, energia de HSV, energia de H, energia de S, energia de V, média de HAS que é  $(hue + saturação)/2$  e entropia de HAS. Foram geradas redes neurais com diferentes quantidades de neurônios de entradas, de um até cinco. Para cada quantidade, foram feitas todas as combinações possíveis de atributos. Portanto, foram avaliadas todas as combinações de um, dois, três, até cinco atributos como entradas das redes, totalizando 27.890 classificadores.

A seleção dos classificadores é dividida em duas fases: na primeira fase, todos os classificadores foram avaliados utilizando apenas os blocos de um frame para o treinamento e os de outro frame similar para a avaliação. Com essa fase foi possível eliminar os classificadores que não alcançaram bom desempenho. Na segunda fase, os classificadores de bom desempenho da primeira fase foram reavaliados com casos bem mais complexos distribuídos entre de quinze frames, onde cinco destes foram usados no treinamento. Essa divisão em duas fases aconteceu para diminuir a quantidade de dados gerados, pois intuitivamente um conjunto de atributos que não conseguiu bons resultados em um caso de teste simples, pode não alcançar bons resultados em um caso bem mais complexo. Depois, utilizando os classificadores selecionados, testamos o sistema de duas formas: sem classificar o céu e determinar a posição da linha do horizonte, processando a imagem toda a cada frame, e classificando o céu e determinando a posição da linha do horizonte a cada 20 frames. Assim, é possível dimensionar a redução do custo computacional de acordo com a posição da linha do horizonte.

#### 4.1. FASE 1

Preliminarmente, todos os classificadores foram avaliados utilizando apenas os elementos de um frame para o treinamento e os de outro frame para a avaliação. Nota-se que esses frames são bem parecidos - Figura 3. Outro detalhe é que não são utilizados todos os elementos dos frames. Como boa parte superior da imagem do frame é composta de céu, a classificação é realizada apenas a partir de uma determinada altura da imagem. O mesmo ocorre na classificação de qual porção da imagem é composta por céu, pois uma boa parte da região inferior não é composta por céu. Isso não afeta na análise de desempenho do classificador, uma vez que a porção do céu ou terra pode ser eliminada com um pré-processamento assim como fez [16]. Isso faz com que sejam utilizados apenas 480 blocos por frame.



(a) Frame utilizado no treinamento. (b) Frame utilizado na avaliação.

Figura 3: Frames usados na Fase 1.

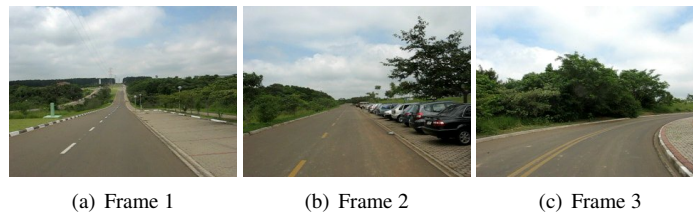


Figura 4: Alguns dos frames usados no treinamento da *Fase 2*.

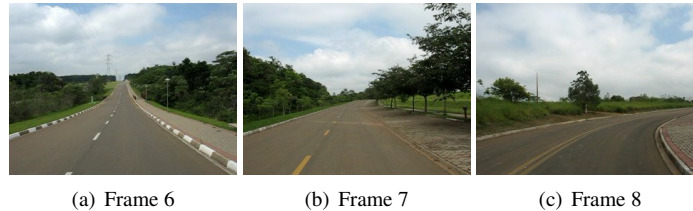


Figura 5: Alguns dos frames restantes usados apenas na etapa de avaliação da *Fase 2*.

Dentre todos os classificadores da região navegável avaliados, cerca de 16.976 deles obtiveram uma taxa de acerto entre 90% e 98%, onde mil deles alcançaram o melhor resultado. O mesmo ocorreu com os classificadores da região composta por céu. É importante notar que essa porcentagem está relacionada com os 480 blocos do frame de avaliação, o que significa que um classificador que obteve uma taxa de acerto de 98% errou na classificação de apenas dez blocos. Apesar dos resultados serem bons, o conjunto de classificadores de bom desempenho é muito diversificado e sua quantidade ainda é intratável. Devido a isso, uma segunda fase de testes foi realizada nos classificadores da região navegável, afim de obter uma avaliação mais criteriosa. Os classificadores da região da imagem composta por céu, tiveram um resultado satisfatório e não foram submetidos à segunda fase de testes.

#### 4.2. FASE 2

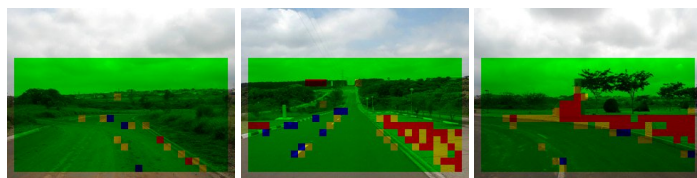
Baseado nos resultados da primeira fase, os classificadores que alcançaram uma taxa de acerto maior ou igual a 90% foram avaliadas com novas condições de rua e céu. Ao invés de treinar apenas com um frame, os classificadores foram treinados com cinco frames bem diferentes e distribuídos ao longo do percurso. Para avaliar o desempenho, foram utilizados quinze frames, dos quais cinco são os frames utilizados no treinamento. Essa forma de avaliação foi escolhida para verificar até que ponto a generalização da rede, uma vez que o aprendizado se faz com vários exemplos de condições diferentes, se mantinha estável. Dentre os frames usados no treinamento pode-se ver curvas, ruas sujas e ruas não muito bem definidas nas bordas - Figura 4.

Dentre todos os classificadores da região navegável avaliados, cerca de 5.967 destes alcançaram uma taxa de acerto maior ou igual a 90% e a maior taxa de acerto foi apenas 93%. Um detalhe importante para se notar é que nessa fase, a análise é mais complexa que na fase anterior, uma vez que a porcentagem de acerto está relacionada a (15 \* 480) blocos. O que significa que o melhor classificador, com 93% de acerto, errou na classificação de cerca de 504 blocos. Os blocos classificados erroneamente podem estar distribuídos entre os quinze frames ou podem estar todos concentrados em um ou dois frames, podendo tornar um classificador uma péssima escolha se não soubermos em qual situação ele não possui resultados razoáveis. Devido a isso, é necessário fazer uma análise visual das respostas dos classificadores, nessa análise são vistos onde e como estão distribuídos os blocos classificados erroneamente. A partir dessa análise, pôde-se inferir condições de uso para cada classificador.

#### 4.3. ANÁLISE 1

Dentre os cinco mil classificadores de bom desempenho, fizemos uma análise visual simplória que pode ajudar a entender as características e limitações de um classificador baseado em redes neurais com atributos de imagem específicos. Para tal, analisamos as cinco melhores configurações de RNAs que atingiram 93% de acerto, enquanto o restante das cinco mil RNAs atingiram menos que isso. A descrição das combinações de entradas das RNAs é dada a seguir:

- RNA 1: média de B, entropia de RGB, entropia de V, variância de S, energia de S.
- RNA 2: média de B, entropia de HSV, entropia de S, entropia de V, energia de S.
- RNA 3: média de R, média de B, média de H, entropia de V, energia de HSV.
- RNA 4: média de R, média de G, média de H, entropia de V, entropia de HAS.
- RNA 5: média de R, média de H, entropia de H, entropia de V ( essa configuração possui apenas quatro atributos).



(a) Análise do Frame 14 com RNA 1. (b) Análise do Frame 1 com RNA 1. (c) Análise do Frame 11 com RNA 1.

Figura 6: Visualização de algumas das respostas das RNAs: mostra classificação correta, falso-negativo, falso-positivo e indeterminado em verde, azul, vermelho e amarelo, respectivamente.

A RNA 1 obteve erros na classificação de uma parte da área do estacionamento (Fig. 6(b)) e calçadas (Fig. 6(c)), obteve erros nas faixas de trânsito e meio fio. Entretanto conseguiu ótimos resultados nas imagens com a rua suja (Fig. 6(a)). De um modo geral, os erros ficaram distribuídos por todos os frames.

A RNA 2 obteve erros na classificação de toda a área do estacionamento e calçadas. Nos exemplos com rua suja, muitos blocos foram classificados com a classe indeterminado - lembrando que indeterminado é considerado erro no cálculo das porcentagens de acerto. O desempenho foi muito bom nos outros casos.

A RNA 3 foi a que obteve melhor precisão na área de estacionamento e calçadas. Diferente das outras RNAs, esta obteve muitos erros na classificação das regiões perto das faixas de trânsito. Além disso, obteve bons resultados nas ruas sujas.

A RNA 4 obteve erros consideráveis na área do estacionamento e calçadas, entretanto os erros foram do tipo indeterminado. A mesma obteve uma das melhores classificações sobre a rua suja, e também obteve erros no meio fio e faixas de trânsito, mas no geral bom desempenho nas demais cenas.

A RNA 5, assim como a RNA 2, obteve erros na classificação de toda a área do estacionamento e calçadas. Mas obteve bom desempenho nas situações restantes. Vale lembrar que essa RNA possui apenas quatro atributos.

De um modo geral, todas as RNAs obtiveram erros nas bordas, meio fio e faixas de trânsito. Todas classificaram corretamente a maior parte da estrada onde o carro deve trafegar. Outra informação que pode ser aproveitada é que essas cinco RNAs possuem vários atributos em comum. Devido a isso, foi feita outra análise, onde foi verificado quais subcombinações de atributos apareceram mais entre os cinco mil classificadores de desempenho maior ou igual a 90%.

#### 4.4. ANÁLISE 2

Com base nos experimentos anteriores, foi realizado uma análise de frequência dos atributos utilizados buscando encontrar fatores em comum entre os classificadores de bom desempenho. Mais especificamente, dentre todos os 5.967 classificadores da região navegável que atingiram uma taxa de acerto maior que 90%, existem configurações de RNAs com uma, duas, três, até cinco atributos como entrada. Então foi contabilizado quantas vezes um subconjunto  $X$  de atributos foi utilizado como entrada. Onde  $X$  é um subconjunto de até quatro elementos dentre os 21 atributos gerados.

Analisando os subconjuntos de quatro elementos, foram encontradas combinações de atributos que alcançaram bom desempenho sempre que utilizadas em conjunto. Tais configurações são mostradas na Tabela 1, essas três combinações apareceram 18 vezes, - que é o valor máximo para subconjuntos de quatro elementos, uma vez que existem apenas 17 atributos restantes para fazerem parte do subconjunto como o quinto elemento e existe a possibilidade de não possuir o quinto elemento - o que significa que é irrelevante o atributo usado na posição do quinto elemento.

Tabela 1: Atributos mais usados.

|   |
|---|
| média de B, média de H, entropia de H, entropia de V      |
| média de H, entropia de H, entropia de V, energia de HSV  |
| média de H, entropia de H, entropia de V, entropia de HAS |

Note que esses subconjuntos possuem (média de H, entropia de H e entropia de V) em comum, que é o subconjunto de 3 elementos que possui a maior frequência dentre os 5 mil classificadores - esse subconjunto aparece 146 vezes. Com base nos atributos da Tabela 1, foram reanalisados todos os classificadores que possuem alguma combinação desses atributos. Os classificadores reanalisados são mostradas na Tabela 2, onde as colunas AT são atributos, a coluna AVE é a média de acerto de dez execuções diferentes, a coluna SD é o desvio padrão dessa média, AFP é média da taxa de erro por frame e a coluna SDF é seu desvio padrão. A Tabela 2 está ordenada pela coluna AVE.

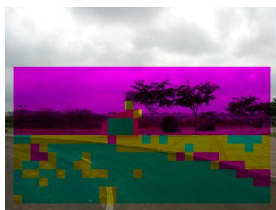
A partir dos resultados mostrados na Tabela 2, note que as dez melhores combinações por AVE - com alta taxa de acerto -, também estão entre as dez melhores combinações que obtiveram a menor média na taxa de erro por frame, e pequeno desvio padrão entre eles. A melhor RNA possui 6.93% de erro por frame e um baixo desvio padrão. Em outras palavras, a melhor RNA errou menos que 10% em todos os frames em que foi testada, numericamente é um bom resultado.

Analisando visualmente as respostas das melhores RNAs, as que contém o atributo (média de B) tendem a classificar os estacionamentos e calçadas como indeterminado - (Fig. 7(a))- enquanto as RNAs sem o atributo (média de B) classificam

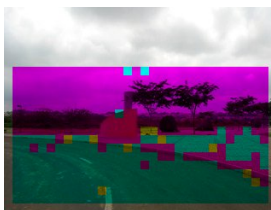


Tabela 2: Avaliação dos atributos mais usados. ( $P(x)$  = energia de  $x$ ), ( $E(x)$  = entropia de  $x$ ), ( $M(x)$  = média de  $x$ )

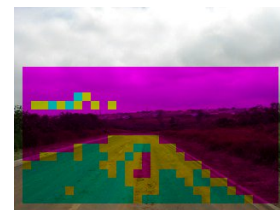
| AT1    | AT2  | AT3  | AT4  | AVE  | SD   | AFP   | SDF   |
|--------|------|------|------|------|------|-------|-------|
| M(B)   | M(H) | E(H) | E(V) | 92   | 0.94 | 6.93  | 3.32  |
| E(HAS) | M(H) | E(H) | E(V) | 91.7 | 0.82 | 7.74  | 3.68  |
| M(B)   |      | E(H) | E(V) | 91.6 | 0.7  | 7.28  | 4.49  |
| M(B)   |      |      | E(V) | 91.5 | 0.53 | 7.38  | 4.59  |
| P(HSV) | M(H) | E(H) | E(V) | 91.2 | 0.63 | 7.74  | 3.62  |
|        | M(H) | E(H) | E(V) | 91.0 | 0.47 | 8.18  | 3.21  |
| E(HAS) | M(H) |      | E(V) | 91.0 | 0.0  | 8.54  | 3.59  |
| P(HSV) | M(H) |      | E(V) | 90.5 | 0.53 | 8.72  | 3.59  |
|        | M(H) |      | E(V) | 90.3 | 0.48 | 8.56  | 3.73  |
| M(B)   | M(H) |      | E(V) | 90.2 | 1.32 | 7.59  | 4.3   |
| M(B)   | M(H) | E(H) |      | 89.8 | 0.63 | 9.04  | 3.8   |
|        | M(H) | E(H) |      | 89.0 | 0.67 | 10.0  | 3.11  |
| E(HAS) | M(H) |      |      | 89.0 | 0.0  | 11.93 | 4.08  |
| P(HSV) | M(H) | E(H) |      | 89.0 | 0.0  | 11.22 | 2.98  |
| M(B)   | M(H) |      |      | 88.9 | 0.99 | 10.17 | 3.74  |
| M(B)   |      | E(H) |      | 88.8 | 0.79 | 10.97 | 4.75  |
| E(HAS) | M(H) | E(H) |      | 87.7 | 7.27 | 9.29  | 4.25  |
| P(HSV) | M(H) |      |      | 87.5 | 0.85 | 13.31 | 4.41  |
| E(HAS) |      | E(H) | E(V) | 78.3 | 2.71 | 19.86 | 5.95  |
| E(HAS) |      | E(H) |      | 74.5 | 0.85 | 23.82 | 6.94  |
| E(HAS) |      |      | E(V) | 70.6 | 0.7  | 28.15 | 5.45  |
| P(HSV) |      | E(H) | E(V) | 67.5 | 0.53 | 31.47 | 8.59  |
|        |      | E(H) | E(V) | 65.7 | 0.48 | 33.54 | 10.38 |
| P(HSV) |      |      | E(V) | 63.5 | 0.71 | 36.92 | 8.86  |
| P(HSV) |      | E(H) |      | 59.3 | 2.11 | 42.22 | 7.24  |



(a) Calçada como indeterminado. Com média de B.



(b) Calçada como navegável. Sem média de B.



(c) Erros na rua com sujeira. Com média de B não classifica rua suja como navegável.

Figura 7: Visualização das respostas das RNAs: mostra blocos classificados como não-navegável - em magenta -, como navegável - em ciano - e indeterminado - em amarelo.

como navegável - (Fig. 7(b)) - (classificação errada). Entretanto as RNAs com (média de B) ficam mais sensíveis a sujeira da rua, obtendo classificações piores na rua suja - (Fig. 7(c)).

Uma conclusão que pode ser notada é que o subconjunto (média de H e entropia de V) possui uma boa performance, porque ele aparece oito vezes entre os dez melhores da Tabela 2. Também pode-se ver que quando esse subconjunto é combinado com outro atributo, ele ganha um pequeno aumento na sua taxa de classificação. Devido a esse fato foram reanalisadas todas as RNAs de até três elementos que contém esse subconjunto. A Tabela 3 mostra a mesma análise da Tabela 2 para essas novas RNAs.

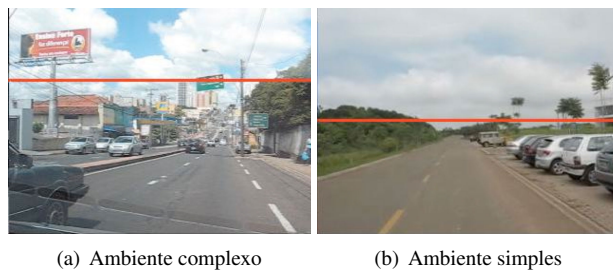
Tabela 3: Avaliação de M(H) e E(V). ( $P(x)$  = energia de  $x$ ), ( $E(x)$  = entropia de  $x$ ), ( $M(x)$  = média de  $x$ ), ( $var.$  = variância)

|    | AT 1 | AT 2 | AT 3   | AVE  | SD   | AFP   | SDF  |
|----|------|------|--------|------|------|-------|------|
| 1  | M(H) | E(V) | M(G)   | 91.3 | 1.1  | 7.54  | 4.0  |
| 2  | M(H) | E(V) | M(B)   | 91.1 | 0.83 | 7.33  | 4.21 |
| 3  | M(H) | E(V) | E(H)   | 91.1 | 0.3  | 7.54  | 3.37 |
| 4  | M(H) | E(V) | M(V)   | 90.9 | 1.58 | 6.86  | 4.63 |
| 5  | M(H) | E(V) | E(S)   | 90.8 | 0.4  | 7.63  | 3.87 |
| 6  | M(H) | E(V) | S var. | 90.8 | 0.6  | 7.82  | 4.43 |
| 7  | M(H) | E(V) | E(HAS) | 90.8 | 0.4  | 9.0   | 3.32 |
| 8  | M(H) | E(V) | H var. | 90.7 | 0.9  | 7.9   | 4.01 |
| 9  | M(H) | E(V) | HSE(V) | 90.7 | 0.46 | 8.14  | 3.93 |
| 10 | M(H) | E(V) | M(S)   | 90.6 | 1.02 | 8.24  | 2.69 |
| 11 | M(H) | E(V) | E(RGB) | 90.6 | 0.49 | 8.38  | 3.96 |
| 12 | M(H) | E(V) | P(HSV) | 90.6 | 0.49 | 8.58  | 3.85 |
| 13 | M(H) | E(V) | P(RGB) | 90.4 | 0.49 | 8.51  | 3.8  |
| 14 | M(H) | E(V) | P(H)   | 90.4 | 0.49 | 8.64  | 3.74 |
| 15 | M(H) | E(V) | M(R)   | 90.4 | 1.5  | 9.49  | 6.16 |
| 16 | M(H) | E(V) |        | 90.3 | 0.46 | 8.46  | 3.89 |
| 17 | M(H) | E(V) | P(V)   | 90.3 | 0.64 | 8.74  | 3.76 |
| 18 | M(H) | E(V) | P(S)   | 90.2 | 0.4  | 8.78  | 4.01 |
| 19 | M(H) | E(V) | M(HAS) | 89.8 | 0.6  | 11.01 | 6.29 |
| 20 | M(H) | E(V) | V var. | 88.0 | 7.01 | 7.86  | 4.38 |

A partir dos resultados mostrados na Tabela 3 pode-se notar que os melhores estão entre os seis primeiros, tanto na média geral como na análise por frame. Assim, pode-se concluir que (média de H e entropia de V) são bons atributos para serem usados na classificação de cenas de estrada.

#### 4.5. CONSIDERAÇÕES - ANÁLISE 1 e 2

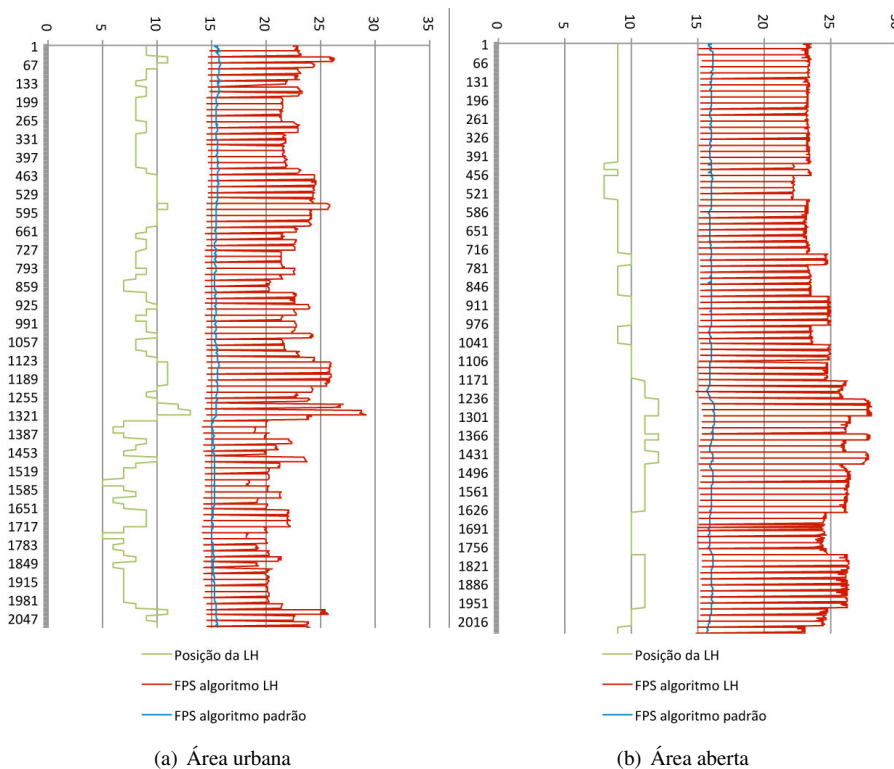
Com base nos experimentos e suas respectivas análises, pode-se concluir que diversos classificadores possuem taxa de acertos semelhante mas suas respostas podem ser diferentes. Isso acontece pela diferença das combinações de atributos avaliadas. Além disso, a análise 2 mostrou que é possível criar classificadores com uma taxa de acerto mínima de 90% se determinados atributos forem utilizados.



(a) Ambiente complexo

(b) Ambiente simples

Figura 8: Ambientes utilizados nos testes de desempenho de um sistema de tempo real, com a posição da linha do horizonte destacada



(a) Área urbana

(b) Área aberta

Figura 9: Desempenho de acordo com a posição da linha do horizonte

Com o método de classificação sendo baseado em blocos, a quantidade de dados a serem processados fica muitas vezes menor dependendo do valor de  $K$  - neste trabalho o ganho de desempenho foi de cem vezes -, logo o processo de classificação se torna muito mais rápido. Graças ao excelente desempenho, pode-se utilizar alguns classificadores baseados em RNAs diferentes e combinar suas respostas para gerar uma classificação final mais robusta e precisa sem uma perda muito grande no desempenho, mantendo, ainda, seu uso em sistemas de tempo real.

#### 4.6. IMPACTO DA DETERMINAÇÃO DA POSIÇÃO DA LINHA DO HORIZONTE

Para avaliar o impacto do desempenho do sistema ao utilizar a determinação da posição da linha do horizonte para reduzir a região da imagem a ser classificada quanto à navegabilidade, avaliamos seu desempenho em dois ambientes distintos. Foram feitos dois testes, e um mesmo teste foi executado duas vezes: a primeira fazendo a classificação da região composta por céu, e a outra apenas processando a imagem toda a cada frame.

O primeiro teste foi executado utilizando um vídeo capturado em um trecho urbano, (Fig. 8(a)), e o segundo em um ambiente aberto, relativamente mais simples (Fig. 8(b)). Em todos os testes a posição da linha do horizonte se mostrou consistente com o valor esperado durante a maior parte do tempo, e não impactou no resultado final da classificação da região navegável.

Nos gráficos 9(a), referente ao teste no ambiente complexo, e 9(b), referente ao ambiente simples, podemos observar que a taxa de frames por segundo (FPS) depende diretamente da posição da linha do horizonte. É notável a diferença de desempenho das duas abordagens. Em média, o ganho de desempenho utilizando a posição da linha do horizonte para reduzir a região da imagem a ser classificada como navegável é de 65%.

## 5. CONCLUSÃO

Navegação autônoma é uma das várias habilidades que um robô móvel ou veículo inteligente deve possuir. Este artigo aborda os problemas de identificação de vias navegáveis e da estimativa da posição da linha do horizonte baseado em visão computacional utilizando redes neurais artificiais. Diferentes combinações de atributos de imagem foram avaliadas em imagens de ambientes reais.

De maneira geral, os resultados foram satisfatórios, uma vez que muitos classificadores obtiveram bons resultados. A região da imagem que contém a rua foi a que menos apresentou erros, o que torna possível uma navegação baseada nestas respostas. A partir disso foi possível descobrir uma relação entre alguns atributos e determinar um grau de contribuição para estes. A classificação da região da imagem composta pelo céu apresentou pouquíssimos erros e se comportou de forma consistente em todos os testes. A utilização da estimativa da posição da linha do horizonte reduziu os ruídos apresentados na identificação da região navegável e o custo de processamento do processo.

Como trabalho futuro, pretende-se determinar alguns classificadores e combinar suas respostas por meio de média ou outras medidas. Desta forma também espera-se conseguir uma classificação ainda mais robusta e precisa. Além disso, pretende-se combinar essas informações com informações de um sensor de distância (laser). E validar o sistema com mais cenários urbanos em condições diferentes.

## 6. AGRADECIMENTOS

Este trabalho contou com o apoio de CNPq e FAPESP por meio do INCT-SEC (Instituto Nacional de Ciência e Tecnologia - Sistemas Embarcados Críticos - Brasil) processo 573963/2008-9e 08/57870-9.

## Referências

- [1] C. Tan, T. hong, T. Chang and M. Shneier, editors. *Color Model-Based Real-Time Learning for Road Following*. IEEE Intelligent Transportation Systems Conference, 2006.
- [2] D. Kristensen. “Autonomous Road Following”. Master’s thesis, KTH Numerical Analysis and Computer Science, 2006.
- [3] E. D. Dickmanns. “Vehicles capable of dynamic vision: a new breed of technical beings?”. *Artif. Intell.*, vol. 103, no. 1-2, pp. 49–76, 1998.
- [4] C. Thorpe, M. Hebert, T. Kanade and S. Shafer. *Vision and navigation for the Carnegie-Mellon Navlab*, vol. 10, no. 1, pp. 362 – 373, May 1988.
- [5] C. Jung and C. Kelber. “A lane departure warning system using lateral offset with uncalibrated camera”. In *Intelligent Transportation Systems, 2005. Proceedings. 2005 IEEE*, pp. 102–107, Sept. 2005.
- [6] Y. He, H. Wang and B. Zhang. “Color-based road detection in urban traffic scenes”. *Intelligent Transportation Systems, IEEE Transactions on*, vol. 5, no. 4, pp. 309–318, Dec. 2004.
- [7] A. Broggi and S. Bertè. “Vision-Based Road Detection in Automotive Systems: A Real-Time Expectation-Driven Approach”. *Journal of Artificial Intelligence Research*, vol. 3, pp. 325–348, 1995.
- [8] J. Crisman and C. Thorpe. *SCARF: A Color Vision System that Tracks Roads and Intersections*, vol. 9, no. 1, pp. 49 – 58, February 1993.
- [9] C. Rasmussen, editor. *Grouping Dominant Orientations for Ill-Structured Road Following*. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004.
- [10] P. Chaturvedi, A. A. Malcolm and J. Ibañez Guzman, editors. *Real-Time Road Following In Natural Terrain*. IEEE Conference on Cybernetics and Intelligent Systems, 2004.
- [11] O. Ramstrom and H. Christensen, editors. *A Method for Following Unmarked Roads*. IEEE, 2006.
- [12] P. Jeong and S. Nedeveschi, editors. *Local Difference Probability (LDP)-Based Environment Adaptive Algorithm for Unmanned Ground Vehicle*. IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, 2006.
- [13] Y. Guo, V. Gerasimov and G. Poulton, editors. *Vision-Based Drivable Surface Detection in Autonomous Ground Vehicles*. IEEE International Conference on Intelligent Robots and Systems, 2006.
- [14] C. E. Shannon. “A mathematical theory of communication”. *Bell system technical journal*, vol. 27, 1948.
- [15] G. Bradski and A. Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library*. O’Reilly, Cambridge, MA, 2008.
- [16] J. Lee, C. D. C. III, S. Kim and J. Kim, editors. *Road Following in an Unstructured Desert Environment using Monocular Color Vision as Applied to the DARPA Grand Challenge*. International Conference on Control, Automation and Systems, 2005.