

Modelos de Markov Ocultos Aplicados na Identificação de Comportamento de Serpentes

Wesley Nunes Gonçalves, Jonathan de Andrade Silva, Hemerson Pistori

¹Grupo de Pesquisa em Engenharia e Computação
Universidade Católica Dom Bosco
Av. Tamandaré, 6000, Jardim Seminário
79117-900 Campo Grande, MS

Abstract.

Resumo.

1. Introdução

As serpentes foram sugeridas como um dos principais grupos de animais para avaliar hipóteses ecológicas e evolucionárias [Rivas and Burghardt 2005]. Além disso, esse grupo é um interessante e importante formador de venenos que serve como base para o desenvolvimento de diversos fármacos. Devido a esses motivos, o uso destes animais na pesquisa comportamental aumentou gradativamente nos últimos anos, principalmente em questões sobre a formação de venenos e o habitat natural.

O habitat para répteis, em particular serpentes, foi ainda pouco explorado na área científica. A criação de cativeiros mais adequados ao habitat das serpentes pode ser auxiliado pela identificação de comportamentos em experimentos científicos, pois na natureza identificar comportamentos e manipular serpentes pode ser muito complexo. O comportamento das serpentes é influenciado por diversos fatores, tal como temperatura, radiação solar, umidade e a estação. A habilidade de identificar e prever a atividade apresentada por uma serpente pode oferecer uma compreensão melhor de seu comportamento exposta aos diferentes fatores e situações, bem como, realizar pesquisas mais confiáveis de novos fármacos constituídos do veneno das serpentes. O veneno das serpentes é um importante componente no desenvolvimento de novos fármacos para seres humanos. Os venenos são usados frequentemente como base para o desenvolvimento de diversos fármacos, tais como, fármacos para controle da hipertensão, analgésicos, anticoagulantes, entre outros.

A identificação comportamental de animais é realizada na maioria das vezes de forma manual, com o experimentador observando o animal e anotando a ocorrência de algum padrão de comportamento considerado importante. Essa observação pode ser realizada com um investimento relativamente baixo, mas pode ter sua precisão comprometida com a fadiga ou distração do observador. Quando existe mais de um observador, pode ocorrer, classificações diferentes para o mesmo comportamento apresentado pelo animal. Na observação automatizada é possível registrar os comportamentos de forma confiável e consistente durante longos períodos de tempo. Os comportamentos consistem em posturas e mudança de posturas no tempo, devido a essas mudanças, identificar os comportamentos é considerado complexo e um dos fatores que limitam o desenvolvimento de ferramentas automáticas. Entretanto, a observação automática é extremamente útil quando o nível de atividade do animal é baixo, o que ocorre com as serpentes, e são necessárias

diversas horas ou dias seguidos para o desenvolvimento da pesquisa. A observação automática, é ainda particularmente apropriada quando há necessidade de medidas exatas, como a velocidade e a distância utilizada para medir o nível de atividade da serpente, em que o observador humano é incapaz de estimar exatamente [Spink et al. 2001].

O objetivo deste trabalho é avaliar os modelos ocultos de Markov na identificação automática do comportamento de bote realizado por uma serpente. O modelo foi treinado com 20 sequências, totalizando 1000 quadros, e a avaliado através de 10 sequências, totalizando 500 quadros. A segmentação dos quadros foi realizada de forma semi-automática através do algoritmo máquina de vetores de suporte[] com atributos do espaço RGB. A melhor porcentagem de classificação e tempo de execução foi alcançada através da variação dos parâmetros dos modelos ocultos de Markov, como o número de estados e quantização. Além disso, três maneiras de inicialização dos HMMs foram avaliadas. O comportamento de bote foi inferido com exatidão de 96 % pelo HMM com XX estados, XX bits de quantização e XX tipo de inicialização.

Este artigo está estruturado em oito seções. A Seção 2 apresenta alguns trabalhos correlatos que utilizam os modelos de Markov ocultos em diversas aplicações. Na Seção 3, é descrita a técnica de segmentação utilizada para a identificação das serpentes. Na próxima seção, são apresentados os extratores de características das imagens segmentadas. Na Seção 5, os principais conceitos e algoritmos dos modelos de Markov ocultos são descritos. Na Seção 6, os experimentos realizados são mostrados com maiores detalhes. A Seção 7 apresenta os resultados obtidos seguido de uma análise. Finalmente, na Seção 8 são mostradas a conclusão e os trabalhos futuros.

2. Trabalhos Correlatos

Os modelos de Markov ocultos têm sido amplamente utilizados em diversas áreas, principalmente em sistemas para o reconhecimento de voz [Rabiner 1990], reconhecimento de comportamentos [Leo et al. 1999] e textos manuscritos [Hu et al. 1996]. Em [Hornegger et al. 1994] o HMM é aplicado ao reconhecimento de objetos 2D em imagens. O HMM, juntamente com as características invariantes do contorno adquiridas dos objetos, foram testados em quatro diferentes objetos. Para cada objeto, um HMM é estimado com um conjunto de cinquenta imagens de treinamento. A classificação é realizada utilizando dez imagens para cada objeto, resultando em uma taxa de acerto de 75%.

Starner e Pentland [Starner and Pentland 1995] descrevem um sistema para reconhecimento de sentenças da língua americana de sinais utilizando HMM. A taxa de reconhecimento é de 99.2% para palavras, sem explicitamente modelar os dedos das mãos. No entanto, o conjunto de característica mostrou-se limitado, o sistema é treinado para esperar certos gestos em certas posições espaciais na imagem devido à utilização das posições não normalizadas das mãos. Para corrigir esse problema Starner e Pentland [SP95] sugerem a utilização de posições normalizadas no vetor de características. Uma nova técnica para o reconhecimento de textos é apresentada em [Aas et al. 1995]. As características são extraídas de uma imagem em tons de cinza e um HMM é modelado para cada caracter. Durante o reconhecimento, a mais provável combinação de modelos é encontrada para cada palavra, pelo uso de programação dinâmica.

Em [Nefian and Hayes 1998] é descrito um HMM para reconhecimento e detecção de faces. A imagem contendo a face é dividida em cinco blocos (cabelo, testa,

olhos, nariz e boca), onde cada bloco é representado como um estado no HMM. Os vetores de características são obtidos de cada bloco utilizando o coeficiente da transformada de Karhunen-Loeve. A detecção é feita calculando-se a probabilidade dos dados de entrada com um dado modelo de face. Para detectar falsos alarmes, apenas probabilidades maiores que um limiar são reconhecidas como faces.

Os modelos ocultos de Markov são bastante utilizados para descrever uma sequência de padrões, como comportamentos. Em [Leo et al. 1999] os comportamentos de seres humanos são identificados. Esses comportamentos são atividades legais e ilegais realizadas em um sítio arqueológico. Para a identificação desses comportamentos, as imagens são segmentadas utilizando a detecção de movimento e um pós-processamento para remoção de sombras. Em seguida, a postura realizada pelo ser humano é identificada utilizando histogramas e medida de similaridade baseada na distância de Manhattan. O reconhecimento dos comportamentos é realizado pelos modelos de Markov ocultos, onde os estados são as posturas. Os experimentos foram realizados na identificação de quatro comportamentos com uma porcentagem média de classificação correta de 86,87%.

Em [Feldman and Balch 2003] um sistema de classificação de comportamentos dos animais é apresentado. Esse sistema usa uma combinação de HMM e kNN para treinamento dos movimentos. O sistema foi avaliado em diversas trajetórias de abelhas extraídas de uma sequência de vídeo de 15 minutos. Os movimentos foram etiquetados manualmente e pelo sistema. O sistema desenvolvido foi capaz de etiquetar movimentos com uma exatidão de 81,5%. O reconhecimento de complexos movimentos de objetos utilizando HMM é apresentado em [Bashir et al. 2005]. O modelo é construído baseado em análise de componentes principais (PCA). Os estados do HMM são representados por Misturas de Gaussianas e o modelo é automaticamente construído dos dados de treinamento. Os experimentos são executados em duas séries de dados. Uma série são imagens contendo sinais da língua australiana de sinais (ASL) contendo 207 trajetórias que descrevem três palavras. A outra série contém 108 trajetórias de atletas executando esportes. Os resultados chegaram a uma taxa de acerto de 90%.

Os diversos sistemas que fazem o uso de modelos ocultos de Markov para o reconhecimento de comportamentos apresentam bons resultados. Entretanto, a taxa de reconhecimento depende muito do conjunto de características visuais utilizadas e do número de estados do modelo. Em [Montero and Sucar 2004] é realizada a análise de diferentes características no reconhecimento de gestos visuais produzidos pelas mãos. Os resultados obtidos mostram uma alta variação na taxa de reconhecimento, devido a combinação de diferentes características. As taxas de reconhecimento obtiveram uma variação de 50% até 97%. O melhor desempenho foi obtido pelo uso da magnitude e orientação em coordenadas polares e dez estados no modelo.

Além do HMM, outros métodos foram utilizados para o reconhecimento de comportamentos. Em [Rousseau et al. 1998] é realizada a classificação do comportamento de um camundongo por uma rede neural. Foram classificados sete comportamentos como: caminhar, sentar, elevar-se. Após o treinamento, a rede classificou corretamente 84,85% dos comportamentos. Segundo Rousseau [Rousseau et al. 1998] essa porcentagem de acerto é comparável à de um observador humano.

3. Momentos de Imagem

As imagens são constituídas de diversos objetos, pertencentes ao plano de fundo e ao plano de interesse. Os objetos do plano de interesse são, geralmente, os que possuem maior importância para os sistemas de visão computacional. Esses objetos são constituídos de um conjunto de *pixels* representando a sua forma, cor e textura. O extrator de momentos de imagem utilizado neste trabalho descrevendo a área do objeto considerando seu conteúdo interno.

Os momentos [?] de ordem $p + q$ sobre uma função contínua 2D são calculados de acordo com a Equação 1.

$$M_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx dy \quad (1)$$

Uma imagem pode ser visualizada como uma função 2D discreta I , onde a intensidade de cada *pixel* é indexada como $I(x, y)$. A Equação 2 é a derivação da Equação 1 de momentos no conceito de imagens.

$$M_{pq} = \sum_{x=1}^{width} \sum_{y=1}^{height} x^p y^q I(x, y) \quad (2)$$

Para a representação de um objeto através de momentos são considerados apenas *pixels* pertencentes ao objeto. Através dos momentos de imagens, aplicados a um objeto, é possível derivar algumas propriedades, como por exemplo, a área do objeto é representada pelo momento M_{00} e o centro de massa \bar{x} e \bar{y} é representado pelos momentos $\frac{M_{10}}{M_{00}}$, $\frac{M_{01}}{M_{00}}$, respectivamente.

Além dos momentos regulares, é possível derivar os momentos centrais para imagens, definida na Equação 3.

$$u_{pq} = \sum_{x=1}^{width} \sum_{y=1}^{height} (x - \bar{x})^p (y - \bar{y})^q I(x, y) \quad (3)$$

As propriedades derivadas dos momentos centrais neste trabalho são a variância em X e Y (Equação 4), a direção do objeto (Equação 5) e a excentricidade (Equação 6)

$$\sigma_x^2 = \frac{u_{20}}{m_{00}}, \quad \sigma_y^2 = \frac{u_{02}}{m_{00}} \quad (4)$$

$$\theta = \frac{u_{02} - u_{20} - 2u_{11} + \lambda}{u_{02} - u_{20} + 2u_{11} - \lambda} \quad (5)$$

$$e^2 = \frac{u_{20} + u_{02} + \lambda}{u_{20} + u_{02} - \lambda} \quad (6)$$

$$\lambda = \sqrt{(u_{20} - u_{02})^2 + 4u_{11}^2} \quad (7)$$

Na Figura 1 é ilustrada a aplicação dos momentos em uma imagem contendo uma serpente. A imagem foi dividida em quatro quadrantes com a mesma área, em seguida, para cada quadrante são extraídos o centro de massa e o desvio, ilustrados como eixos nas cores azul, laranja e verde. Além desses parâmetros, para a imagem como um todo, é extraída a elipse que representa o objeto.

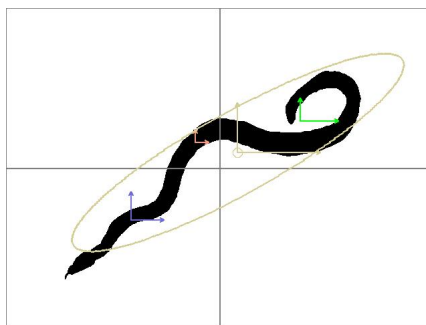


Figure 1. Aplicação dos momentos de imagem.

4. Modelos de Markov Ocultos

Os modelos ocultos de Markov são compostos por um processo estocástico duplo. O primeiro processo é descrito por um conjunto de estados não diretamente observáveis, por isso, o nome desse modelo recebe a palavra oculto. Para os modelos ocultos de Markov de primeira ordem, a probabilidade de transição entre os estados depende apenas do estado anterior, representada através de uma matriz de transição $A = \{a_{ij}\}$ descrita na Equação 8.

$$a_{ij} = P(q_{t+1} = S_j | q_t = S_i) \quad 1 \leq i, j \leq N \quad (8)$$

sendo N o número de estados, q_t o estado atual e $S = \{S_1, S_2, S_3, \dots, S_N\}$ o conjunto de estados do modelo.

Em algumas aplicações, os estados não são diretamente observados, com isso, cada estado é observado através de um segundo processo estocástico. A seqüência de observações é representada por um conjunto $O = O_1, O_2, O_3, \dots, O_T$, onde cada elemento O_t representa um símbolo do conjunto finito de símbolos $V = v_1, v_2, \dots, v_M$ sendo M o número de símbolos. A probabilidade de emissão de símbolos em um estado j é definida na matriz $B = \{b_j(k)\}$ da Equação 9.

$$b_j(k) = P(O_t = v_k | q_t = S_j) \quad 1 \leq j \leq N, \quad 1 \leq k \leq M \quad (9)$$

A probabilidade inicial de cada estado é representada por um conjunto $\pi = \{\pi_i\}$ mostrada na Equação 10.

$$\pi_i = P(q_1 = S_i) \quad 1 \leq i \leq N, \quad \text{com} \quad \sum_{i=1}^N \pi_i = 1 \quad (10)$$

Para a utilização dos modelos ocultos de Markov em aplicações do mundo real, é necessário a resolução de três problemas básicos. A resolução desses problemas, conhecidos como problema de avaliação, decodificação e aprendizagem, foram extensivamente descritas em diversos trabalhos na literatura [?, ?, ?, ?].

4.1. Problema de Avaliação

O primeiro problema envolve o cálculo da probabilidade de uma seqüência observada $O = O_1, O_2, \dots, O_T$ ser produzida por um modelo $\lambda = (A, B, \pi)$. Esse problema é conhecido como um problema de avaliação e podemos visualizar a solução desse problema como uma probabilidade que mensura a relação entre a seqüência observada e um modelo. Um dos procedimentos que resolve esse problema eficientemente, baseado na técnica de programação dinâmica, é conhecido como *Forward-Backward*. Esse procedimento define uma variável $\alpha_t(j)$, mostrada na Equação 11, utilizando-a para o cálculo da probabilidade $P(O|\lambda)$.

$$\alpha_t(j) = P(O_1, O_2, O_3, \dots, O_t, q_t = S_j | \lambda) \quad (11)$$

A variável $\alpha_t(j)$ define a probabilidade das observações parciais até um tempo t e estado S_j dado um modelo λ . Essa variável é calculada recursivamente através das Equações 12, 13 e 14.

$$\alpha_1(j) = \pi_j b_j(O_1), \quad 1 \leq j \leq N. \quad (12)$$

A Equação 12 inicializa a variável $\alpha_t(j)$ com a probabilidade inicial de cada estado e a probabilidade desse estado gerar a primeira observação O_1 .

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(O_{t+1}), \quad 1 \leq t \leq T - 1. \quad (13)$$

A Equação 13 é o principal passo do procedimento. Esse passo calcula, recursivamente, a probabilidade de um estado S_j gerar uma observação O_{t+1} e ser alcançado no tempo $t + 1$ pelos N possíveis estados no tempo t .

$$P(O|\lambda) = \sum_{j=1}^N \alpha_T(j) \quad (14)$$

O último passo realizado pelo procedimento *Forward-Backward* é dado na Equação 14. Esse passo calcula a probabilidade $P(O|\lambda)$ através da soma das variáveis $\alpha_T(i)$ finais. Esse procedimento, comparado com a solução acima, é mais eficiente, envolvendo cálculos na ordem de N^2T .

A variável $\beta_t(i)$ é calculada de forma similar a variável $\alpha_t(i)$. Essa variável é definida na Equação 15 calculando a probabilidade de observações parciais de $t + 1$ até T dado o estado S_i no tempo t e um modelo λ .

$$\beta_t(i) = P(O_{t+1}, O_{t+2}, O_{t+3}, \dots, O_T | q_t = S_i, \lambda) \quad (15)$$

A variável $\beta_t(i)$ poder ser calculada recursivamente através das Equações 16, 17 e 18. A Equação 16 realiza a inicialização dos parâmetros do procedimento, a Equação 17 realiza o passo da indução do algoritmo e finalmente a Equação 18 realiza o cálculo de $P(O|\lambda)$.

$$\beta_T(i) = 1, \quad 1 \leq i \leq N. \quad (16)$$

$$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(O_{t+1}) \beta_{t+1}(j), \quad t = T-1, T-2, T-3, \dots, 1 \quad 1 \leq i \leq N. \quad (17)$$

$$P(O|\lambda) = \sum_{i=1}^N \pi_i b_i(O_1) \beta_1(i) \quad (18)$$

Para o problema de avaliação, não é necessário que as duas variáveis α e β sejam calculadas. Entretanto, para solucionar os outros problemas, como o problema de treinamento, as duas variáveis são necessárias.

4.2. Problema de Decodificação

O segundo problema, conhecido como problema de decodificação, busca a melhor seqüência de estados $Q = q_1, q_2, \dots, q_T$ que representa a seqüência de observações $O = O_1, O_2, \dots, O_T$ baseado em um critério ótimo. A solução existente para resolver esse problema é através do algoritmo baseado em programação dinâmica, conhecido como algoritmo de Viterbi. O algoritmo de Viterbi escolhe a melhor seqüência de estados dado as observações e pode ser definido através da Equação 19.

$$\delta_t(i) = \max_{q_1, q_2, \dots, q_t} P(q_1, q_2, \dots, q_t = i, O_1, O_2, \dots, O_t | \lambda). \quad (19)$$

A variável $\delta_t(i)$ define a maior probabilidade da seqüência de estados q_1, q_2, \dots, q_t gerar cada elemento da seqüência de observação O_1, O_2, \dots, O_t percorrendo um caminho ótimo ao longo do modelo até um estado S_i no tempo t , considerando as probabilidades de transições entre os estados e as observações. Dessa forma, por indução a solução do problema pode ser descrita através da Equação 20.

$$\delta_{t+1}(j) = [\max_i \delta_t(i) a_{ij}] b_j(O_{t+1}). \quad (20)$$

Para maximizar a Equação 20, é necessário armazenar a seqüência percorrida em um caminho no modelo no tempo t em um estado j . Esse percurso ao longo do modelo é armazenado em um vetor $\psi_t(j)$. Assim definida as variáveis, o processo de execução do algoritmo é definido em três passos. A Equação 21 realiza o passo de inicialização das variáveis, em seguida, na Equação 23 é realizado o passo de recursão do algoritmo, no qual é calculado e armazenado o melhor caminho pelo modelo. Finalmente a Equação 25 realiza a finalização do algoritmo recuperando o caminho pelo modelo por *backtracking*.

$$\delta_1(j) = \pi_j b_j(O_1), \quad 1 \leq j \leq N. \quad (21)$$

$$\psi_1(j) = 0. \quad (22)$$

$$\delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] b_j(O_t), \quad 2 \leq t \leq T \quad 1 \leq j \leq N. \quad (23)$$

$$\psi_t(j) = \arg \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}], \quad 2 \leq t \leq T \quad 1 \leq j \leq N. \quad (24)$$

$$P^* = \max_{1 \leq j \leq N} [\delta_T(j)]. \quad (25)$$

$$q_T^* = \arg \max_{1 \leq j \leq N} [\delta_T(j)]. \quad (26)$$

$$q_t^* = \psi_{t+1}(q_{t+1}^*), \quad t = T-1, T-2, \dots, 1. \quad (27)$$

4.3. Problema da Aprendizagem

O terceiro problema, considerado um dos mais difíceis do HMM, consiste em determinar um método para ajustar os parâmetros do modelo $\lambda = (A, B, \pi)$, que maximiza localmente $P(O|\lambda)$. Nos modelos ocultos de Markov, os estados não são diretamente observáveis, com isso, não é possível identificar a qual estado pertence uma certa observação. Devido a esse problema, não é possível utilizar uma fórmula ou algoritmo analítico para realizar a estimação dos parâmetros. Neste caso é necessário utilizar métodos iterativos, como o algoritmo de *Baum-Welch* (especialização do algoritmo EM).

O algoritmo de *Baum-Welch* possui um método iterativo para a re-estimação dos parâmetros baseado no método de estimação por máxima verossimilhança. A idéia básica desse algoritmo é realizar a estimação através do número esperado de transições do estado S_i para o estado S_j e o número de emissão do símbolo O_t no estado S_i . A estimação desses números é baseada nos valores dos parâmetros atuais do modelo e na seqüência de observação do treinamento.

Para iniciar a estimação dos parâmetros do modelo é necessário definir a probabilidade de transitar de um estado S_i no tempo t para outro estado S_j no tempo $t+1$ dado a seqüência de observação de treinamento para o modelo λ , definido pela Equação 28.

$$\xi_t(i, j) = P(q_t = S_i, q_{t+1} = S_j | O, \lambda). \quad (28)$$

As probabilidades de transição entre os estados S_i e S_j são calculadas através dos métodos *Forward-Backward*, discutidos na seção ???. A Equação 28 pode ser descrita através das variáveis do algoritmo *Forward-Backward*, na forma da Equação 29.

$$\xi_t(i, j) = \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{P(O|\lambda)}. \quad (29)$$

$$\xi_t(i, j) = \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}. \quad (30)$$

A variável $\gamma_t(i)$ já foi definida na Seção ?? como a probabilidade de ocorrência do estado S_i . Essa variável pode ser reescrita em função de $\xi_t(i, j)$ através da Equação 31.

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i, j). \quad (31)$$

Ao realizar a soma de $\gamma_t(i)$ obtem-se uma medida esperada do número de vezes que o estado S_i foi visitado no tempo t conforme a Equação 32, e ao somar $\xi_t(i, j)$ obtem-se o número de transições ocorridas do estado S_i para o estado S_j , conforme a Equação 33.

$$\sum_{t=1}^{T-1} \gamma_t(i). \quad (32)$$

$$\sum_{t=1}^{T-1} \xi_t(i, j). \quad (33)$$

Com base nas equações acima, que realizam a contagem de eventos ocorridos, pode-se obter um método de estimar os parâmetros do modelo $\lambda = (A, B, \pi)$. Dessa forma, as equações 34, 35 e 36, representam respectivamente a frequência de eventos ocorridos no estado S_i no tempo 1, a média do número de transições ocorridas do estado S_i para o estado S_j e finalmente a média do número de símbolos observados no estado S_j .

$$\bar{\pi}_i = \gamma_1(i). \quad (34)$$

$$\bar{a}_{i,j} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}. \quad (35)$$

$$\bar{b}_j(O_t) = \frac{\sum_{t=1, o_t=v_k}^{T-1} \gamma_t(i)}{\sum_{t=1}^{T-1} \gamma_t(i)}. \quad (36)$$

Os novos parâmetros do modelo $\bar{\lambda} = (\bar{A}, \bar{B}, \bar{\pi})$, foram re-estimados através das equações 34, 35 e 36. O processo de re-estimação é realizado até que sejam atendido um dos critérios de parada. O critério de parada é definido quando a máxima verossimilhança é atingida, com $\bar{\lambda}$. Além desse critério, é possível estabelecer um número fixo de iterações.

5. Experimentos

Os experimentos com os modelos ocultos de Markov foram realizados com 30 sequências de imagens representando ou não o comportamento de bote. As imagens, totalizando 1500, foram capturadas com uma camera TRENDNET TV-IP301W com resolução de 640 x 480 *pixels* de uma visão área do catifeiro, simulando um habitat natural (Figura

??). Em seguida, os quadros foram segmentados através do algoritmo máquina de vetores de suporte (*Support Vector Machine* - SVM). Para realizar a segmentação, foram coletadas amostras positivas, correspondentes a pele das serpentes, e amostras negativas, correspondentes a diferentes texturas do fundo. O modelo construído pela SVM, foi baseado em atributos RGB extraídos de uma região centrada em um pixel (x, y) mantendo a ordem estabelecida da região.

Após a segmentação, são adquiridos vetores multidimensionais com valores contínuos obtidos através dos momentos de imagens. Como os experimentos foram realizados com os HMMs discretos, foi necessário um passo de discretização dos valores extraídos. Essa discretização foi realizada através do algoritmo *Vector Quantization LBG* []. Esse algoritmo discretiza os atributos através de um número de núcleos selecionado, com número incrementado a cada . Com isso, foi gerado um HMM para modelar o comportamento de bote e outro HMM para modelar o restante das sequências de imagens. Todos os experimentos foram realizados em um computador com processador P4 2.8GHz, 512MB de memória RAM e sistema operacional Fedora Core 5.

Para avaliar a eficiência dos HMMs e encontrar a melhor porcentagem de classificação correta, a principal etapa dos experimentos foi a variação dos parâmetros. Para o número de estados essa variação foi realizada de XX até XX, o número de bits utilizado na quantização dos atributos variou de XX a XX e o número de iterações do algoritmo de aprendizagem variou de XX a XX. A inicialização dos HMMs foi realizada de forma equiprovável, com probabilidade igual para todos os valores da matriz A e B, aleatória e através do algoritmo K-Means [] aplicado no conjunto de treinamento. O número de núcleos do K-Means é de acordo com o número de bits utilizado na discretização.

6. Resultado e Análise

Os resultados são apresentados nas Figuras ??, ?? e ??. Na Figura ?? o gráfico apresenta a mudança na porcentagem de classificação correta pelo número de estados do HMM. Apenas o resultado médio para as configurações de parâmetros é plotada. Com XX estados os HMMs alcançaram uma classificação correta de XX%, entretanto, o tempo de execução (Figura ??) apresenta um alto custo. Embora com XX estados os HMMs obtiveram uma porcentagem um pouco inferior de XX %, o tempo de execução foi consideravelmente menor com relação a XX estados.

A relação entre o número de bits na quantização e a porcentagem de classificação é apresentada na Figura ??. Com valor de XX bits na discretização, a porcentagem de acerto foi de XX%. Essa porcentagem se manteve até XX bits. Na Figura ?? é avaliado o número de iterações utilizado no algoritmo *Baum-Welch*. A partir de XX iterações, a porcentagem continuou em torno de XX% com tempo de execução (Figura ??) aumentando.

7. Conclusão e Trabalhos Futuros

References

- Aas, K., Eikvil, L., and Andersen, T. (1995). Text recognition from grey level images using hidden Markov models. *Lecture Notes in Computer Science*, 970:503–508.
- Bashir, F., Khokhar, A., and Schonfeld, D. (2005). Hmm-based motion recognition system using segmented pca. *Proc. IEEE International Conference on Image Processing*.

- Feldman, A. and Balch, T. (2003). Automatic identification of bee movement. Technical report, Georgia Institute of Technology, Atlanta, Georgia 30332, USA.
- Hornegger, J., Niemann, H., Paulus, D., and Schlottke, G. (1994). Object recognition using Hidden Markov Models. In Gelsema, E. S. and Kanal, L. N., editors, *Pattern Recognition in Practice IV: Multiple Paradigms, Comparative Studies and Hybrid Systems*, volume 16, pages 37–44, Amsterdam. Elsevier.
- Hu, J., Brown, M. K., and Turin, W. (1996). Hmm based on-line handwriting recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(10):1039–1045.
- Leo, M., D’Orazio, T., and Spagnolo, P. (1999). *Human Activity Recognition for Automatic Visual Surveillance of Wide Areas*. Academic Press, 1 edition.
- Montero, J. A. and Sucar, L. E. (2004). Feature selection for visual gesture recognition using hidden markov models. *ENC*, 0:196–203.
- Nefian, A. V. and Hayes, M. H. (1998). Face detection and recognition using hidden markov models. In *ICIP (1)*, pages 141–145.
- Rabiner, L. R. (1990). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77:257–286.
- Rivas, J. A. and Burghardt, G. M. (2005). Snake mating systems, behavior, and evolution: The revisionary implications of recent findings. *Journal of Comparative Psychology*, 119(4):447–454.
- Rousseau, J. B. I., van Lochem, P. B. A., Melder, W., Florêncio, C. C., and Gispen, W. H. (1998). Classification of rat behavior by a neural network. *2 International Conference on Methods and Techniques in Behavioral Research*, pages 18–21.
- Spink, A. J., Tegelenbosch, R. A., Buma, M. O., and Noldus, L. P. (2001). The ethovision video tracking system – a tool for behavioral phenotyping of transgenic mice. *Physiol Behav*, 73(5):731–744.
- Starner, T. and Pentland, A. (1995). Visual recognition of american sign language using hidden markov models. Technical Report Master’s Thesis, MIT, Program in Media Arts & Sciences, Program in Media Arts - MIT, Cambridge, USA.