

O Uso de Redes Neurais Artificiais na Detecção de Pele em Imagens Digitais visando o Reconhecimento de Gestos

João Ricardo Bittencourt (UNISINOS)
jrbitt@netu.unisinos.br

Fernando Santos Osório (UNISINOS)
osorio@exatas.unisinos.br

Resumo. O objetivo deste trabalho é apresentar o uso das Redes Neurais Artificiais na detecção de pele em imagens digitais. Com isto visamos criar uma aplicação para o reconhecimento de gestos capturados por uma WebCam de forma que possamos aumentar a interação homem-máquina. Constatou-se através da experimentação que com a utilização de bases de dados simples para o treinamento de filtros adaptativos baseados em Redes Neurais Artificiais e com o uso de *thresholds* adequados podemos obter resultados significativos adequados ao tipo de aplicação proposto.

Palavras-chave: Inteligência Artificial, Processamento de Imagens, Aprendizado de Máquinas, Redes Neurais Artificiais, Detecção de Pele.

1 Introdução

Atualmente existe um grande número de pesquisas sendo desenvolvidas sobre o reconhecimento de faces e de mãos em imagens digitais. Para efetuar esta identificação uma abordagem inicial amplamente utilizada consiste em identificar na imagem quais os pixels representam pele.

A pele é formada basicamente por uma substância denominada melanina [1]. As pessoas que possuem mais desta substância possuem a pele mais escura, ao contrário das pessoas que possuem menos melanina. A partir desta propriedade é possível afirmar que a pele possui uma cromaticidade bastante característica que facilita a identificação deste elemento nas imagens. Muitos trabalhos que estão sendo desenvolvidos buscam resolver este problema através de modelos estatísticos, mas tais trabalhos encontram várias dificuldades ao se tentar criar um modelo genérico de cromaticidade. Como é bem sabido, o reconhecimento de pele pode ser fortemente influenciado pelas diferentes raças, pela iluminação do ambiente, pelo tipo de câmera utilizada, entre outros fatores. Uma alternativa para contornar este problema, e que se destaca neste tipo de aplicações é o desenvolvimento de sistemas que possam facilmente ser adaptados e personalizados de modo a realizar um reconhecimento de um determinado tom de pele, uma cromaticidade específica. Outro grande problema desta detecção refere-se aos objetos que podem apresentar pontos com a mesma cromaticidade da pele e acabam sendo detectados erroneamente.

As aplicações para detecção de peles são inúmeras, mas podemos destacar algumas delas. Primeiramente pode ser usada para reconhecimento de gestos. Gesticular é uma das formas mais intuitivas de comunicação humana e por esta razão é um elemento bastante importante para efetuar a interação entre homem e máquina [2], seja interagindo com um computador ou com um robô. Outra aplicação refere-se a identificação de imagens pornográficas [3], visando um melhor controle deste tipo de conteúdo que é difundido livremente na Internet e permitindo assim uma restrição automática do acesso a este tipo de imagens por parte de crianças ou em instituições onde seu acesso seja inadequado e/ou indesejado. A grande maioria destas imagens possui uma grande extensão de pele que é uma característica importante para identificação destas imagens.

Uma das formas de efetuar este tipo de detecção é usando Redes Neurais Artificiais (RNA). Bittencourt e Osório [4] propuseram uma metodologia para efetuar processamento de imagens através de filtros adaptativos usando RNAs, especificamente através do uso do algoritmo de Backpropagation. Estes filtros são capazes de serem treinados de forma automática para se transformarem em detectores de pele. A característica mais importante destes filtros é a possibilidade de adaptação para o reconhecimento de determinado tipo de pele. Além disso, podem ser treinados de forma a operar de modo mais preciso de acordo com determinado sistema de aquisição de imagens, como por exemplo, WebCams. Justamente esta capacidade de adaptação para identificação de pele nos permite criar mais facilmente sistemas que possam interagir com seres humanos.

Os experimentos que foram realizados neste trabalho visam principalmente a implementação do detector de pele em aplicações de reconhecimento de gestos capturados através de uma WebCam. É importante destacar que os gestos que deverão ser reconhecidos são extremamente simples, tais como dedo polegar apontando para cima e dedo indicador apontando para cima e para baixo.

Para isto este trabalho está organizado da seguinte forma: na seção 2 alguns trabalhos relacionados à detecção de pele serão discutidos, principalmente com a intenção de verificar como esta detecção é efetuada e o modelo de cor utilizado; na seção 3 será apresentado o aplicativo *Neuron Color* [4] que permite criar bases de exemplos para o treinamento das RNAs, criar filtros neurais e manipular imagens com estes filtros; na seção 4 será apresentada a metodologia de criação destes filtros neurais; na seção 5 será discutida uma série de casos aplicados a detecção de pele usando filtros neurais e buscando verificar a eficiência destes métodos variando aspectos referentes às bases de exemplos, o uso do contexto e a aplicação de um pós-processamento para melhoria da imagem detectada pelos filtros neurais; na seção 6 os resultados serão discutidos; e por último na seção 7 serão feitas as considerações finais e serão apresentadas possibilidades de trabalhos futuros.

2 Trabalhos Relacionados

Conforme foi dito na introdução uma série de pesquisas estão sendo desenvolvidas, principalmente para o reconhecimento de gestos [2] e a criação de filtros de imagens pornográficas [5] [3].

Em geral estas pesquisas usam o modelo YCrCb [1] [6] ou RGB normalizado [2] [7] ao contrário de utilizar a tripla RGB. O principal motivo de usar estes modelos é que a questão da iluminação é amenizada e a cromaticidade é valorizada. Mas o sistema RGB tradicional também é usado por Jones e Regh [8] principalmente pelo fato de usarem imagens capturadas na Internet que estão representadas neste sistema de cor. Utilizar um outro modelo de cor implicaria em efetuar um pré-processamento.

Em todos os filtros desenvolvidos neste trabalho foi utilizado o modelo RGB, pelas seguintes razões: (a) as imagens são capturadas pela WebCam no sistema RGB. Um pré-processamento para conversão de cores reduz o desempenho do filtro em consequência comprometendo o processamento em tempo real requerido para o reconhecimento de gestos; (b) o *Neuron Color* foi projetado para gerar as bases de exemplos e manipular imagens neste modelo de cor; (c) todos os experimentos que foram executados em trabalhos anteriores usando esta ferramenta tiveram resultados significativos e usaram o modelo RGB. De uma certa forma, neste trabalho procuramos criar os filtros neurais de uma forma mais padronizada, dentro das mesmas características dos trabalhos anteriores. Mas evidentemente que a hipótese de utilizar o sistema YCbCr, RGB normalizado ou outro modelo de cores em trabalhos futuros não está descartada.

Quanto às técnicas de implementação para detecção de pele encontramos referências de trabalhos que se utilizam de: misturas de Gaussianas [2] [7], Teorema de Bayes [2] [6], além de simples *thresholds* [1] [6]. Em Phung, Chai e Bouzerdoum [1] as Redes Neurais são utilizadas e o algoritmo utilizado para o aprendizado foi o Levenberg-Marquardt. Em Brown, Craw e Lewthwaite [9] os mapas de Kohonen [10] são utilizados para detectar pele em sistemas de tempo real. Evidentemente que existem inúmeras propostas de soluções para o problema de detecção de pele, mas o enfoque do nosso trabalho é o uso de RNAs, mais especificamente de redes de múltiplas camadas usando algoritmo de Backpropagation ou alguma variação deste para efetuar o treinamento.

Quanto às bases de exemplos, em geral são grandes contendo uma grande diversidade de exemplos (tonalidades de pele). Nestes trabalhos procura-se encontrar um “modelo universal” [1] capaz de detectar qualquer tipo de pele. No caso dos filtros de imagens pornográficas evidentemente que existe esta necessidade de generalização, mas no reconhecimento de gestos pode existir a possibilidade de efetuar um treinamento a priori de um filtro capaz de detectar a cromaticidade de pele de um determinado usuário (filtro personalizado).

Um último ponto que precisa ser considerado nos trabalhos existentes é que em todos os trabalhos referidos anteriormente nesta seção as abordagens utilizadas consideram somente um único pixel na detecção de pele, ou seja, classificar um

pixel como pele ou não-pele não depende de sua vizinhança, somente de sua representação cromática. Na seção 5 veremos se considerar estes pixels vizinhos pode contribuir na eficiência da detecção.

3 Neuron Color

Antes de explicar como funciona a criação dos filtros neurais é necessário apresentar o sistema denominado *Neuron Color* [4] que foi construído para viabilizar o processamento de imagens inteligente usando RNAs.

O *Neuron Color*¹ é uma ferramenta desenvolvida utilizando a linguagem Java da Sun Microsystems e é livremente distribuída para comunidade através da licença GNU GPL [11]. Para o desenvolvimento deste trabalho foram usadas duas aplicações deste sistema: o *Neuron Color Factory* que permite a criação dos filtros neurais e a geração de bases de exemplos para efetuar o treinamento das RNAs; e o *Neuron Color Viewer* que é responsável pela visualização de imagens, aplicação dos filtros neurais e armazenamento das imagens transformadas.

É importante destacar que o *Neuron Color* necessita lidar com algoritmos de RNA. Para isto foi construído o *ANNeF* (Artificial Neural Networks Framework) [12] um conjunto de classes genéricas que oferece a implementação de algoritmos de RNA, incluindo o Backpropagation. Desta forma o *Neuron Color* utiliza o ANNeF para modelar as RNAs dos filtros neurais.

Na seção seguinte será descrito o método usado para construir os filtros neurais e por conseqüência mais detalhes sobre o *Neuron Color* serão apresentados.

4 Criação dos Filtros Neurais

O objetivo desta seção é apresentar como os filtros neurais usados na detecção de pele foram criados. Podemos dizer que o uso dos filtros neurais implica em quatro momentos: (1) Criação das bases de exemplos para treinamento e teste; (2) Execução do aprendizado da RNA; (3) Manipulação das imagens aplicando os filtros neurais; (4) Aplicação de um pós-processamento (opcional).

4.1 Criação das bases de exemplos para treinamento e teste

Cria-se em qualquer ferramenta de edição gráfica duas imagens ($train_T$ e $test_T$) a partir de duas imagens originais distintas ($train_O$ e $test_O$). As imagens são transformadas através da manipulação do usuário que buscou aplicar um tratamento nestas imagens correspondente ao filtro que está sendo considerado. No caso de detecção de pele é efetuada uma segmentação manual, tudo que é pele é marcado com a cor branca (1) e tudo que é não-pele é marcado de preto (0).

¹ Disponível em: <http://www.inf.unisinos.br/~jrbitt/nc> Acesso em: 30 ago. 2002

Um par de imagens (original, transformada) é usado para treino e o outro para teste de generalização da RNA, como mostra a Figura 1.

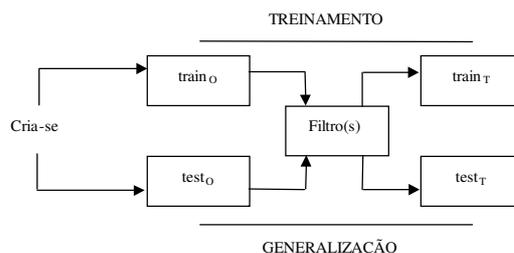


Figura 1 – Esquema da criação das imagens para o treinamento das RNAs

Após as imagens terem sido criadas utiliza-se o *Neuron Color Factory*, para a criação de um novo filtro neural e das bases de dados. Nesta etapa três parâmetros principais devem ser definidos:

Uso de Contexto (vizinhança): para efetuar a transformação de imagens pode ser considerada uma matriz de contexto (pixels vizinhos). Desta forma o processamento de imagens deixa de ser ponto a ponto e passa a considerar uma região em torno do pixel que está sendo processado.

Entradas e Saídas: estas definições dependem das imagens de entrada e das imagens transformadas. Elas que determinam o número de neurônios de entrada e de saída de uma rede neural. No caso do reconhecimento de pele sempre são usadas como entrada imagens coloridas e são geradas imagens em preto e branco. Desta forma as RNAs basicamente são criadas com um único neurônio de saída e o número de neurônios de entrada irá depender do uso do contexto. Como sempre utilizamos matrizes de contexto com as dimensões 7x7 tivemos RNAs com 147 neurônios de entrada (49 neurônios para cada canal RGB). Quando não foi usado contexto, as RNAs ficaram com 3 neurônios de entrada.

Após o filtro ter sido parametrizado as bases de exemplos podem ser criadas, ou seja, cada pixel das imagens será expresso em termos dos valores numéricos de seus componentes RGB, de sua vizinhança e do valor da saída esperada (aprendizado supervisionado). Duas bases devem ser geradas, uma para o treinamento da RNA e outra para o teste de generalização.

4.2 Execução do aprendizado da RNA

Ao término da criação dos arquivos, deve-se usar uma outra ferramenta para realizar o aprendizado da RNA. Nesta etapa pode ser usado o *NevProp* [13], ou o *ANNeF* [12] que possui o *BackpropSimulator*, como sendo o simulador de RNAs.

Inicia-se então o processo de aprendizagem, seja com o *NevProp* ou com o *BackpropSimulator*. Durante esta fase de treinamento o usuário deve intervir diretamente no que diz respeito aos parâmetros utilizados no aprendizado, ao

término da simulação (adição de épocas) e a verificação da queda do erro global. Este processo de ajuste de parâmetros é repetido até obter uma boa generalização

Com posse do arquivo de pesos, gerado pelo *NevProp* ao final do aprendizado, utiliza-se mais uma vez o *Neuron Color Factory*, para associar a rede neural obtida ao filtro desenvolvido.

4.3 Manipulação das imagens aplicando os filtros neurais

Finalizada esta etapa podemos visualizar qualquer imagem no formato GIF, JPEG, TIFF, PNG e BMP [14] utilizando o *Neuron Color Viewer*. Através desta aplicação é possível utilizar os filtros neurais e armazenar em disco as imagens transformadas. Nesta aplicação é possível verificar visualmente a qualidade do resultado obtido pela aprendizagem.

4.4 Aplicação de técnicas de pós-processamento

O uso de um algoritmo de pós-processamento pode ajudar no incremento da qualidade das imagens processadas pelos filtros neurais. No caso destes experimentos de detecção de pele foram utilizados um binarizador das imagens e um eliminador de pixels isolados. O primeiro utiliza um *threshold* pré-determinado pelo usuário que determina que qualquer valor abaixo deste limiar é preto e qualquer valor acima é branco. Já o eliminador de pixels isolados considera uma matriz 5x5, um limite de dispersão também definido pelo usuário e uma imagem já binarizada. Supondo um limite de dispersão de 0,5, significa que o pixel central da matriz somente será branco se mais de 50% da sua vizinhança também for branco, caso contrário o pixel é definido como preto.

Nos experimentos descritos na seção seguinte será verificado se ocorre um aumento da qualidade da detecção de pele usando estas duas técnicas de pós-processamento.

5 Experimentações

5.1 Pré-definições

O objetivo principal destas experimentações é verificar se os filtros neurais são eficientes na detecção de pele considerando a aplicação de reconhecimento de gestos. Além disso, serão considerados a influência das bases de exemplos e o uso de uma matriz de contexto na detecção de pele.

Dois medidas de desempenho foram propostas por Caetano e Barone [7] e serão consideradas nestes experimentos: a taxa de detecção correta (*TDC*) e a taxa de falsa detecção (*TFD*), sendo definidos como:

- **TDC**: número de pixels detectados como pele e que realmente são pele dividido pelo número total de pixels que representam pele;
- **TFD**: número de pixels detectados como pele e que realmente não são pele dividido pelo número total de pixels que representam pele.

5.1.1 Geração das bases de exemplos

Para estes experimentos serão consideradas duas bases, uma com uma diversidade de exemplos de pele e não-pele e uma outra base mais simples que consiste da captura de uma imagem por uma WebCam. O modelo utilizado nestas experimentações é uma WebCam Go da Creative Inc.

A razão de utilizar dois tipos de bases é comparar a questão de usar um modelo genérico de reconhecimento de pele com um outro treinamento mais simplificado e dirigido para determinada tonalidade de pele.

As bases com alta diversidade foram criadas a partir de um “mosaico” com 72 imagens rotuladas como pele e 72 imagens rotuladas como não pele. A dimensão de cada uma destas imagens é 30x30 pixels. A imagem obtida consiste de 64.800 pixels, sendo que 50% são pixels rotulados como pele e a outra metade como não-pele. Duas imagens foram criadas uma para treino e outra para o teste de generalização, conforme pode ser visto na Figura 2.

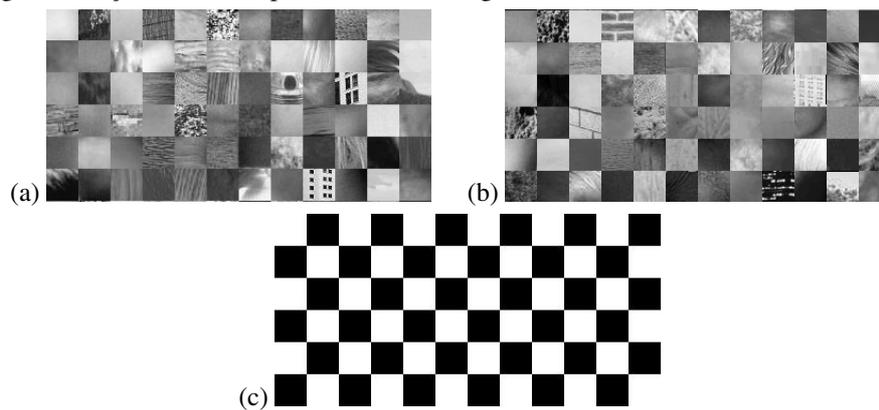


Figura 2 – Imagens com alta diversidade de pixels. Estas imagens foram redimensionadas e originalmente (a) e (b) são coloridas. A imagem (a) foi usada para treino, (b) para teste e (c) é a binarização que classifica os pixels como pele (cor branca) e não-pele (cor preta).

Todas as imagens usadas para compor o mosaico foram obtidas na Internet. Como exemplos de pele, temos imagens de diferentes raças (branco, negro e amarelo), diferentes idades (criança, adulto e idoso) e diferentes intensidades de luz (clara, normal e escura). Já as imagens que não são pele são formadas por exemplos de rocha, tijolo, grama, folhas, cimento, céu, nuvem, água, terra, fogo, cabelo louro, cabelo castanho, cabelo preto, cabelo ruivo, madeira, metal e imagens de cidades e paisagens rurais.

As bases de exemplos foram criadas escolhendo aleatoriamente 5.000 exemplos (pixels). Das Figuras 2(a) e 2(c) foi criada uma amostra para treinamento e das Figuras 2(b) e 2(c) foi criada a base para teste.

Já as bases de exemplos mais simples utilizaram duas imagens capturadas pela WebCam com dimensões de 320x240 pixels, totalizando 76.800 pixels. Veja

Figura 3. A partir das imagens (Figura 3(a) e Figura 3(c)) foram criadas duas imagens segmentadas manualmente que detectam pele (Figura 3(b) e Figura 3(d)).

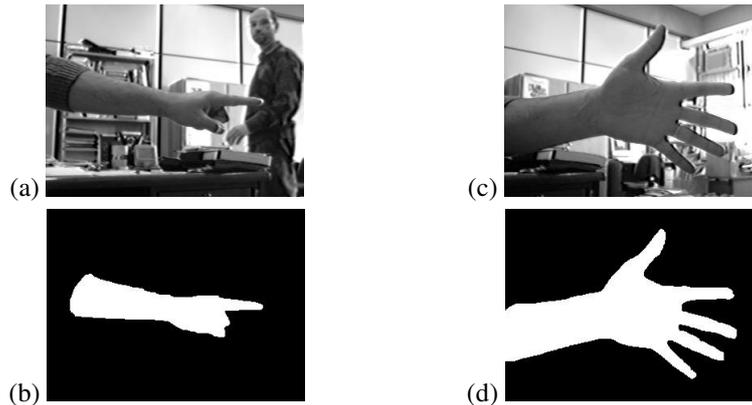


Figura 3 – Imagens simplificadas. Estas imagens foram redimensionadas e as imagens (a) e (c) originalmente são coloridas. As imagens (a) e (b) foram usadas para efetuar os treinos das RNAs e as imagens (c) e (d) foram usadas para teste.

Da mesma forma que as bases com maior diversidade de exemplos, foram criadas as bases para treino e teste simplificadas com 5.000 exemplos cada uma escolhidos aleatoriamente a partir das imagens da Figura 3.

5.1.2 Uso do Contexto

Foram usadas duas variações: utilizando uma matriz de contexto com as dimensões 7×7 e sem a matriz de contexto. A partir disto quatro bases de exemplos foram criadas: (a) base com diversidade usando contexto; (b) base com diversidade sem usar o contexto; (c) base simples usando contexto; e (d) base simples sem usar o contexto.

5.1.3 Treinamento das RNAs

Para cada uma das bases mencionada na seção 5.1.2 foi executado um processo de treinamento. O software para efetuar o treinamento das RNAs foi o *NevProp* pelo fato deste implementar o algoritmo de Quickprop, uma variante do Backpropagation que permite um treinamento das RNAs mais rápido.

Como o processo de treinamento das RNAs é um processo não determinístico, dependendo principalmente da geração dos pesos iniciais das sinapses artificiais que interconectam os neurônios, foram executadas cinco execuções diferentes, cada uma utilizando uma semente randômica diferente.

Na Tabela 1 estão apresentadas as médias das melhores topologias de RNAs para cada um dos experimentos. A partir deste ponto, adotamos a representação E1, E2, E3 e E4 para cada um dos experimentos, combinando contexto e tipo de base conforme indicado na Tabela 1.

	E1	E2	E3	E4
<i>Topologia</i>	3-12-1	147-15-1	147-0-1	3-1-1
<i>Tipo de Base</i>	Diversa	Diversa	Simples	Simples
<i>Contexto</i>	Não	Sim	Sim	Não
<i>Média do % Acertos</i>	74,412%	74,600%	89,148%	82,384%
<i>Desvio Padrão % Acertos</i>	0,398%	0,559%	0,106%	0,704%
<i>RMS Médio</i>	0,412	0,413	0,273	0,321
<i>Desvio Padrão do RMS</i>	0,001	0,006	0,001	0,002

Tabela 1 – Resultado dos treinamentos das RNAs

É importante destacar que o *NevProp* sempre gera o arquivo de pesos para RNA referente a melhor época, ou seja, aquela que apresentou o menor erro médio para a base usada para teste (*cross-validation*). Evidentemente que os filtros neurais criados utilizam estes arquivos.

5.2 Resultados

Para cada experimento foi escolhido o treinamento que apresentou melhor resultado, ou seja, obteve menor erro médio para sua base de teste. Para bases que foram utilizadas para teste de generalização calculou-se a taxa de detecção correta e a taxa de falsa detecção. Os resultados podem ser vistos resumidamente na Tabela 2.

Para obter estes resultados foi considerado um *threshold* de 0,49, o mesmo considerado durante o treinamento. O maior acerto dos pixels que realmente são pele foi obtido usando uma base de exemplos com grande diversidade e não utilizando contexto, mas em contrapartida este mesmo experimento possui a maior taxa de falsa detecção (experimento E1).

Experimento	Tipo de Base	Uso de Contexto	TDC	TFD
<i>E1</i>	Diversa	Não	90,722%	41,790%
<i>E2</i>	Diversa	Sim	86,094%	34,152%
<i>E3</i>	Simples	Sim	72,344%	10,405%
<i>E4</i>	Simples	Não	48,643%	9,050%

Tabela 2 – O TDC e o TFD de cada um dos experimentos. Uso de *threshold* igual a 0,49.

Observando estes resultados é possível constatar que em caso de bases de exemplos com uma grande diversidade usar o contexto auxilia na redução das classificações errôneas (TFD). Já em bases mais simples o uso do contexto implica em uma melhora na taxa de acertos (TDC).

Na Tabela 2 estão resumidas métricas quantitativas, mas observando a Figura 4 é possível efetuar uma avaliação visual dos experimentos. Neste ponto prevalece a questão da subjetividade, pois a avaliação depende de cada pessoa. Através da observação foi possível constatar que as imagens geradas por E1 e E2 apresentam uma taxa de falsa detecção maior que as imagens geradas por E3 e E4. No caso da

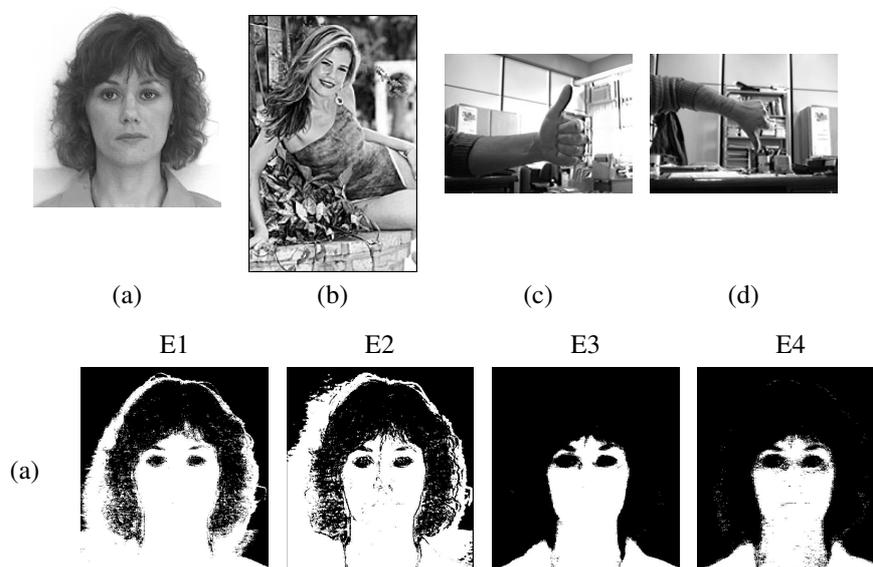
detecção de gestos constatamos que as imagens E3(c) e E3(d) da Figura 4, elas apresentam nitidamente a forma da mão com o polegar apontando para cima e para baixo respectivamente.

É importante destacar que as imagens usadas na Figura 4 são totalmente independentes das imagens usadas para o treinamento e teste de generalização do aprendizado. A Figura 4(a) foi obtida no Departamento de Psicologia da Universidade de Stirling² e a Figura 4(b) no Yagua.Com³. Quanto as Figura 4(c) e Figura 4(d) foram capturadas por nós usando a WebCam.

5.3 Pós-processamento

O objetivo deste pós-processamento é tentar aumentar os índices de acertos de pele e reduzir a os índices de falsa detecção. Entretanto quando o índice de acertos é incrementado (TDC) o índice de falsa detecção também aumenta. O problema está na definição de um ponto (*threshold*) que aumente o índice de acertos sem afetar extremamente o índice de erro.

Os *thresholds* foram escolhidos a partir da análise das curvas ROC e os resultados estão apresentados na Tabela 3. As curvas ROC para cada experimento podem ser observadas na Figura 5.



² Disponível em: <http://pics.psych.stir.ac.uk/> Acesso em: 20 jul. 2002

³ Disponível em: http://modelos.yagua.com/fotos/Modelos_Femeninas_A-L/Luz_Gonzalez/1.html Acesso em: 20 jul 2002



Figura 4 – Exemplo do uso dos filtros neurais para cada um dos experimentos. Todas estas imagens foram redimensionadas e as imagens (a), (b), (c) e (d) originalmente são coloridas. Todas estas detecções de pele estão binarizadas usando um *threshold* de 0,49.

Experimento	Threshold	TDC	TFD
<i>E1</i>	0,6	81,16%	31,32%
<i>E2</i>	0,55	81,062%	28,883%
<i>E3</i>	0,3	83,671%	17,991%
<i>E4</i>	0,3	87,255%	28,431%

Tabela 3 – Uso de um novo *threshold* para cada experimento de forma que aumentasse o TDC ou diminuísse o TFD em relação à Tabela 2.

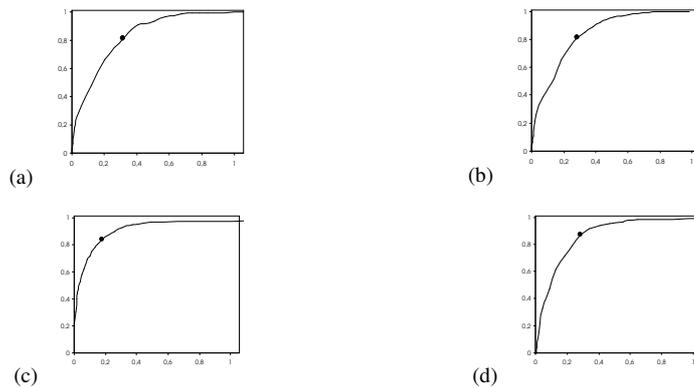


Figura 5 – Curvas ROC. A figura (a), (b), (c) e (d) representam os experimentos E1, E2, E3 e E4 respectivamente. Os pontos pretos são os *thresholds* escolhidos e apresentados na Tabela 3. O eixo x é o TFD e o eixo y é o TDC. O experimento E3 possui a melhor curva.

Na Figura 6 as mesmas imagens da Figura 4 foram novamente binarizadas, mas ao invés de utilizar o *threshold* padrão de 0,49 foram utilizados os novos *thresholds* determinados na Tabela 3. Após este processo ter sido efetuado os pixels isolados foram eliminados. Para isto considerou-se uma matriz 5x5 de pixels e um fator de dispersão de 0,5. Nota-se que principalmente para os experimentos E3 e E4 ocorreu uma grande melhoria visual usando estes dois processamentos.

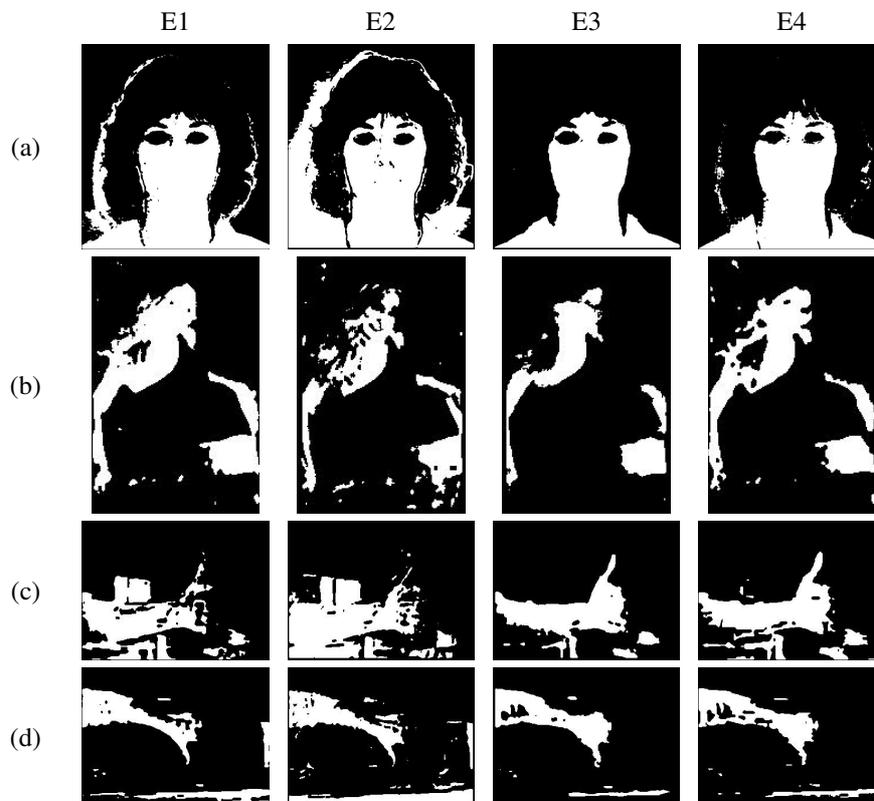


Figura 6 – Nestas imagens ao invés de ser usado o *threshold* 0,49 foi usado o valor correspondente na Tabela 3. Além disso, os pixels isolados foram eliminados.

6 Análise dos Resultados

Através da análise destes dados algumas constatações podem ser feitas. É importante ressaltar que estes dados não são conclusivos, pois somente quatro imagens foram analisadas. E as questões referentes às medidas de desempenho referem-se a uma única imagem usada para teste da generalização. O importante do estudo destes casos é justamente auxiliar na elaboração de estudos futuros mais detalhados.

A primeira constatação refere-se a dificuldade de construir um detector universal de pele. Isto pode ser constatado analisando a Figura 6. Os experimentos E1 e E2 possuem uma grande taxa de erro e os experimentos E3 e E4 possuem os maiores índices de detecção correta. Ambos experimentos utilizaram uma base simples para treinamento. Isto reforça a nossa tese de utilizar um treinamento prévio para detecção de pele, no caso de aplicações de reconhecimento de gestos. Dos quatro experimentos, considerando as bases de exemplos e as topologias de RNAs configuradas, o experimento E3 considerando o *threshold* de 0,3 é que obteve o melhor desempenho.

Quanto ao uso de um pós-processamento percebe-se que realmente os resultados da detecção de pele podem ser melhorados a partir da escolha de um *threshold* adequado. Também a utilização do eliminador de pixels isolados tornou-se uma boa solução pelo fato de eliminar pontos que estão fora de um contexto referente à pele. Evidentemente que usar um pós-processamento, do tipo eliminador de pixels isolados, implica perda de desempenho, fato extremamente prejudicial para aplicações de tempo real, como é o caso do reconhecedor de gestos.

7 Considerações Finais

Ao término deste trabalho é possível concluir que uma abordagem para a detecção de pele visando efetuar um reconhecimento de gestos deverá se utilizar de bases de exemplos mais simples e criar filtros neurais especializados na detecção da cromaticidade da pele de cada usuário. Tanto no aspecto quantitativo, quanto no visual as bases simples apresentaram melhor desempenho.

Quanto ao aspecto do uso do contexto verificou-se na observação destes casos que quando aplicado em bases mais simples apresentou uma melhora significativa no percentual de acertos. Quanto ao uso de um pós-processamento a performance do detector possui um grande incremento. Para isto deve ser determinado corretamente o *threshold* de forma que maximize o índice de acertos sem aumentar significativamente a taxa de erros.

Quanto aos trabalhos futuros pretendemos integrar esta abordagem para detectar pele em uma aplicação capaz de reconhecer gestos. Como podemos observar os melhores resultados aqui apresentados ainda precisam de algum refinamento, para isto seria interessante efetuar testes usando outros modelos de cores, por exemplo, o YCbCr, além de procurar validar estatisticamente estes resultados com amostras maiores. Além disso, é importante destacar que o *Neuron Color* teve um papel fundamental na execução destes experimentos pelo fato de oferecer uma estrutura para criação de bases de exemplos, criação dos filtros e manipulação das imagens.

Referências

- [1] PHUNG, S.L.; CHAI, D; BOUZERDOUM, A. A Universal and Robust Human Skin Color Model using Neural Networks. In: **Proceedings of International Joint**

Conference on IJCNN '01, v. 4, 2844-2849 p. 2001.

- [2] KAPUSCINSKI, T; WYOCKI, M. Hand Gesture Recognition for Man-Machine Interaction. In: **Proceedings of the Second International Workshop on Robot Motion and Control**, 91-96p, 2001.
- [3] JONES, M.J.; REHG, J.M. **Detecting Adult Images**, HP Labs – Cambridge Research Lab, Jun/2002. Disponível em: <<http://crl.research.compaq.com/projects/vision/adult-detection.htm>> Acesso em: 20 jul. 2002.
- [4] BITTENCOURT, J. R.; OSÓRIO, F. S. Adaptive Filters for Image Processing Based on Artificial Neural Networks. In: **Proceedings of SIBGRAPI 2000 - XIII Brazilian Symposium on Computer Graphics and Image Processing**. Los Alamitos: IEEE Computer Society, 2000, p. 336-336
- [5] DRIMBAREAN, A.F. et al. Image Processing Techniques to Detect and Filter Objectionable Images based on Skin Tone and Shape Recognition. In: **ICCE - International Conf. on Consumer Electronics**, p.278 –279. 2001
- [6] PHUNG, S.L; CHAI, D; BOUZERDOUM, A. Skin Color Detection for Face Location in Human-Machine Communications. In: **International Symposium on Signal Processing and its Applications (ISSPA)**, Kuala Lumpur. Ago. 2001.
- [7] CAETANO, T.S; BARONE, D.A.C. A Probabilistic Model for the Human Skin Color. In: **Proceedings of 11th International Conference on Image Analysis and Processing**. 279-283 p. 2001.
- [8] JONES, M.J.; REHG, J.M. **Skin Color Modelling and Detection**, HP Labs – Cambridge Research Lab, Jun/2002. Disponível em: <<http://crl.research.compaq.com/projects/vision/skin-color.htm>> Acesso em: 20 jul. 2002.
- [9] BROWN, D.A; CRAW, I; LEWTHWAITE, J. A SOM Based Approach to Skin Detection with Application in Real Time Systems. In: **Proceedings of the British Machine Vision Conference**, 2001.
- [10] KOHONEN, T. **Self-Organization and Associative Memory**. Springer-Verlag Series in Information Science, 1987.
- [11] Free Software Foundation. **GNU General Public License**, 1991. Disponível em: <<http://www.gnu.org/copyleft/gpl.html>> Acesso em: 20 jul. 2002
- [12] BITTENCOURT, J. R. OSÓRIO, F. ANNeF – Artificial Neural Networks Framework: Uma Solução Software Livre para o Desenvolvimento, Ensino e Pesquisa de Aplicações de Inteligência Artificial Multiplataforma. In: **Anais do II Workshop sobre Software Livre**. Porto Alegre: SBC, p. 13-16, 2001.
- [13] GOODMAN, P.; ROSEN, D.. **NevProp Artificial Neural Network Software with Cross-Validation and Bootstrapped Confidence Intervals**. University of Nevada Center for Biomedical Modeling Research, 2001. Disponível em: <<http://brain.unr.edu/publications/NevProp.zip>> Acesso em: 20 jul. 2002.
- [14] MURRAY, J. D. **Encyclopedia of Graphics File Formats**. 2.ed. O' Reilly & Associates, Inc, 1996. 1.116p.