

# Intelligent Classifier of Banknotes: an approach to Principal Component Analysis and Artificial Neural Networks

Allan K. Lopes e Lais C. R. S. Lopes

Departamento de Informática

Campus Urutaí

Instituto Federal Goiano

Rodovia Geraldo Nascimento, Km 2,5

Urutaí – GO – Brasil

Email: {allankardec.ti, laiscandidolopes}@gmail.com

Fabrizio Soares, Luciana Berretta,

Leandro Oliveira, Anderson Soares e Ronaldo Costa

Instituto de Informática

Universidade Federal de Goiás

Campus Samambaia

Goiânia – GO – Brasil

Email: {fabrizio,luciana,leandroluis,  
anderson,ronaldocosta}@inf.ufg.br

**Abstract**—This paper proposes a method to assist visually impaired people to identify banknotes of Brazilian Real (BRL). The method uses the combination of the technique of Principal Component Analysis (PCA) and artificial neural networks (ANN). The PCA is used in pre-processing the images to reduce the dimensionality of the data. The coefficients obtained with the PCA is used as input to a classifier based on artificial neural network Multilayer Perceptron (MLP). In several experiments, the success rate was 100%. The results obtained with the method were satisfactory. The proposed method has shown promise. It is hoped in the future to recommend its use in helping blind people.

## I. INTRODUÇÃO

O Brasil possui uma população significativa de deficientes visuais. De acordo com o censo realizado em 2010 pelo IBGE<sup>1</sup> cerca de 18,8% da população do Brasil possui algum tipo de dificuldade visual [10]. Embora essa população seja expressiva, ainda há uma carência muito grande de adaptações para deficientes visuais. Os deficientes visuais enfrentam uma série de dificuldades quando interagem com o ambiente à sua volta. A quantidade de informação codificada visualmente torna essa interação ainda mais desafiadora.

Uma dificuldade muito comum de uma pessoa cega é reconhecer o valor das notas de dinheiro que está segurando. No Brasil, a moeda corrente é o real. O real é composto por notas de R\$ 1, 2, 5, 10, 20, 50 e 100. Desde a introdução do real as notas sofreram diversas substituições. Essas substituições foram, em geral, por causa da evolução de mecanismos de segurança, introdução de elementos históricos ou culturais e em alguns casos para inclusão de elementos de apoio aos deficientes visuais. A nota de R\$ 1, por exemplo, de 1994 a 2003 teve 20 versões [2]. Essa nota atualmente está fora de circulação e foi substituída por moedas. Ressalta-se que, o deficiente visual pode levar bastante tempo para se familiarizar com uma nova nota. Infelizmente, cada substituição de nota por um novo modelo contribui para o aumento confusão dos deficientes visuais.

Diversos trabalhos vem sendo pesquisados para reconhecimento de notas. Em [6] foi pesquisado o reconhecimento de notas de dólar utilizando um modelo baseado em descritores obtidos com o método *Speeded Up Robust Features* (SURF). No trabalho de [19] também foi utilizado o reconhecimento de notas de dólar com descritores SURF, porém, com o objetivo de utilização em dispositivos móveis baseados na plataforma Android. [8] desenvolveu um equipamento para reconhecimento de notas de dinheiro que internamente possui os dispositivos BlindNoteReader, NoteTeller 2 e outros. No trabalho de [18] foi realizado o reconhecimento de notas utilizando o algoritmo de aprendizado indutivo RULES-3. Em [1] foi experimentada a utilização de redes neurais artificiais para reconhecimento de notas da moeda Euro. Como entrada foram utilizadas características extraídas de elementos de segurança das notas obtidas com infravermelho. [3] utilizou dados de notas digitalizadas expostas a raios ultravioleta. Os dados foram comparados com mecanismos contra falsificação previstos nas notas.

Neste trabalho é proposto um método para reconhecimento de notas da moeda real. O método usa como entrada imagens digitalizadas das notas de real e as classifica com seu valor correspondente. O modelo proposto combina a análise das componentes principais (PCA) baseada na transformada de Karhunen-Loeve (KLT) e uma rede neural artificial do tipo perceptron multicamadas (MLP). Ressalta-se que o objetivo deste sistema é apenas a identificação do valor das notas e não constatar sua legitimidade.

## II. ANÁLISE DAS COMPONENTES PRINCIPAIS E REDES NEURAIS ARTIFICIAIS

### A. Análise das componentes principais (PCA) e a transformada de Karhunen-Loeve

A análise das componentes principais (PCA) realiza a extração de um conjunto de variáveis chamadas componentes. As componentes são extraídas na ordem da mais explicativa para a menos explicativa [9]. É possível avaliar a exclusão das componentes menos explicativas. Assim, os dados originais são reduzidos a um conjunto significativamente menor. Destaca-se que, embora seja possível reduzir os dados originais

<sup>1</sup>Instituto Brasileiro de Geografia e Estatística

a um conjunto de componentes, a PCA permite reter o máximo possível de informação contida nas variáveis originais.

Um dos tipos de PCA mais utilizados para imagens é a transformada de Karhunen-Loeve (KLT) [11], [14]. Segundo [13], a transformada de Karhunen-Loeve [4] é o método preferido para aproximação de um conjunto de imagens em um sub-espaço de baixa dimensionalidade.

Aplicações com imagens que utilizam a KLT constroem uma matriz  $X = \{X_n \in R^d \mid n = 1, \dots, N\}$ . Nessa matriz  $X$ , cada linha é formada pelas linhas concatenadas das imagens, em que  $d$  é o produto entre a altura e largura de cada imagem. Em seguida, é calculado um vetor média ( $E(X)$ ), conforme a Equação (1). O vetor média do conjunto representa a média de todas as imagens.

$$E(X) = \frac{1}{N} \sum_{n=1}^N X_n \quad (1)$$

Em seguida, é subtraído cada elemento de  $X$  do vetor média ( $E(X)$ ) e obtém-se uma nova matriz, conforme Equação (2). Para isto, cada linha de  $X$  deve ser subtraída do vetor média ( $E(X)$ ). Esta nova matriz contém a variação de cada imagem em torno da média.

$$\bar{X}_n = \{X_n - E(X), n = 1, \dots, N\} \quad (2)$$

Em seguida, calcula-se a matriz de covariância  $M = \bar{X} \times \bar{X}^T$ . Desta nova matriz ( $M$ ) são extraídos os autovetores  $Q$  e autovalores  $V$ . Por fim, é recalculada, a partir da Equação (3), a matriz de autovetores que seja ortogonal e orthonormal. Esta matriz contém é formada pelas componentes principais.

$$Q^* = \frac{\bar{X}^T \times Q \times (V)^{-0.5}}{\sqrt{N-1}} \quad (3)$$

Finalmente, projeta-se a matriz  $\bar{X}$  na matriz transformação  $Q^*$  para se obter uma nova representação das imagens. Pode-se ordenar os autovetores da matriz  $Q^*$  de acordo com a significância dada pela matriz de autovalores  $V$  e assim eliminar os autovetores de menor representatividade.

Para aplicar o PCA em uma imagem de teste é necessário projetá-la na matriz transformação  $Q^*$ . Assim, seja  $x_{teste}$  a matriz que representa a imagem de teste; então  $x_{teste}^T$  será essa matriz concatenada e  $\bar{x}_{teste}^T$  será essa matriz concatenada subtraída da média global  $E(x)$ , conforme a Equação (4).

$$\bar{x}_{teste}^T = x_{teste}^T - E(x) \quad (4)$$

Projeta-se  $\bar{x}_{teste}^T$  na matriz transformação, conforme Equação (5), e utiliza-se uma regra de decisão afim de se classificar a imagem de teste.

$$\bar{x}_{pca} = \bar{x}_{teste}^T \times Q^* \quad (5)$$

## B. Rede Neural artificial Perceptron Multicamadas (MLP)

A rede neural Perceptron Multicamadas (MLP) é composta de uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída. O sinal de entrada propaga-se para frente pela rede, camada por camada [7]. Cada camada é composta por elementos chamados de neurônios artificiais que possuem entradas, uma função de ativação e uma saída. A Figura 1 mostra a MLP utilizada neste trabalho.

Diversas funções de ativação que podem ser utilizadas na rede MLP [7]. A função de ativação utilizada neste trabalho nos neurônios das camadas oculta e de saída foi a tangente hiperbólica, também chamada de tan-sigmóide ou tansig, calculada conforme a Equação (6).

$$\varphi(a) = \tanh(a) = \frac{2}{(1 + \exp(-2a))} - 1 \quad (6)$$

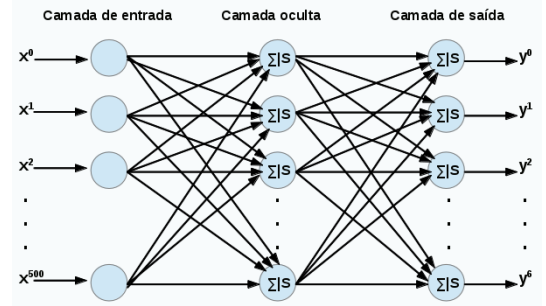


Figura 1. Exemplo da arquitetura de três camadas da rede neural utilizada.

Existe uma variedade de algoritmos para treinamento da MLP. Estes algoritmos são gradiente descendente, o gradiente descendente com momentum, o gradiente conjugado, o quasi-Newton, o Levenberg-Marquardt, dentre outros.

Segundo [5] a técnica de Levenberg-Marquardt é mais eficiente do que a técnica do gradiente descendente convencional. O gradiente descendente é um algoritmo de maior descida e envolve movimentos de pequenos passos abaixo do gradiente local do campo escalar. Uma desvantagem deste método é a possibilidade do gradiente descendente encontrar um mínimo local antes que o mínimo global seja atingido. O algoritmo Levenberg-Marquardt é um aperfeiçoamento do método Gauss-Newton, que é uma variante do método de Newton [12], [15]. O método de Newton usa a informação da derivada parcial de segunda ordem do índice de desempenho utilizado para corrigir os pesos. Isso permite que além da informação do gradiente, seja usada informação sobre a curvatura da superfície do erro.

Neste estudo, todos os códigos foram escritos usando MATLAB. Foi utilizado o Toolbox de Redes Neurais do MATLAB para implementar tanto a MLP quanto, o algoritmo de Treinamento [16].

## III. MODELO PROPOSTO

Para um melhor entendimento da abordagem aplicada na resolução do problema, as principais etapas para a construção do sistema proposto são detalhadas a seguir.

### A. Digitalização das notas para treinamento e teste

Para o treinamento e teste do sistema foram digitalizadas diversas notas de Real. Foram digitalizadas notas de 2, 5, 10, 20, 50 e 100 reais. As notas de 100 reais foram utilizadas em suas duas versões existentes. Ressalta-se que, embora existam diversas versões de cada nota, as versões anteriores são frequentemente recolhidas e a obtenção de amostras é muito dificultado. As notas foram digitalizadas em frente e verso sob um fundo branco. A quantidade de notas digitalizadas de cada tipo foi aleatória. E de acordo com as condições de conservação e semelhança de cada nota, algumas notas não tiveram ambas as faces digitalizadas. A Tabela I apresenta a quantidade de imagens digitalizadas de cada nota.

Tabela I. QUANTITATIVO DE AMOSTAS DIGITALIZADAS DE CADA NOTA

Nota (R\$)	Qtd. Frente	Qtd. Verso	Total
2	38	38	76
5	35	27	62
10	24	34	58
20	31	32	63
50	35	42	77
100 v1	41	47	88
100 v2	35	41	76
<b>Total</b>	<b>239</b>	<b>261</b>	<b>500</b>

Para a captura das imagens das notas foram utilizadas as câmeras RGB primárias de celulares. Os celulares utilizados foram o Samsung modelos Galaxy Y GT S6102B e o Galaxy Pocket S5300, com 3 e 2 megapixels de resolução, respectivamente. As digitalizações foram realizadas manualmente e em diferentes condições de luminosidade, distância e posição. A Figura 2 mostra exemplos das imagens de notas de real capturadas.



Figura 2. Exemplos de imagens de frente e verso das notas de 2, 10 e 50

### B. Pré-processamento

Esta fase é dividida em três passos. Primeiramente, as imagens digitalizadas através da atividade descrita na subseção III-A são convertidas em imagens de 256 tons de cinza e redimensionadas para a resolução de  $300 \times 140$  pixels de largura e altura, respectivamente.

Em seguida, as imagens são separadas em 2 conjuntos: um para treinamento e outro para teste do modelo. O conjunto de

treinamento é composto por 90% das imagens. E o conjunto de teste é composto dos 10% restantes. Cada conjuntos contém, proporcionalmente, representantes de todas as classes. Por exemplo, para a nota de 2 Reais foram digitalizadas 38 imagens da parte da frente e 38 do verso. Deste total, 34 imagens frente e verso foram utilizadas para treinamento e 4 imagens frente e verso foram utilizadas para teste. Ressalta-se que seleção de imagens de cada classe foram selecionadas aleatoriamente para cada conjunto.

Por fim, as imagens são processadas com algoritmo do PCA, descrito na seção II-A. O resultado da aplicação do PCA é utilizado como entrada para a rede MLP.

### C. Arquitetura e inicialização da rede MLP

A classificação das imagens é feita com uma rede neural artificial Perceptron Multicamadas (MLP). A RNA foi projetada com três camadas: uma de entrada, uma oculta e uma de saída. O número de neurônios na camada de entrada é igual ao tamanho dos vetores de entrada que representam as imagens. Essa camada não realiza nenhum processamento, tem a função de apenas distribuir os sinais para a próxima camada de neurônios. Na camada oculta o número de neurônios, no experimento realizado, variou entre 3 e 60 e para a camada de saída foram necessários 6 neurônios, um por nota.

Os pesos e bias das conexões dos neurônios da rede foram inicializados de acordo com a regra de D. Nguyen e B. Widrow [17]. A Figura 1 ilustra a RNA projetada para classificar as imagens. Em ambas as camadas, oculta e de saída, os neurônios utilizam a função de transferência tangente hiperbólica sigmóide.

Para uma maior confiabilidade e rapidez no treinamento da MLP normalizou-se os dados de entrada no intervalo de  $[-1, 1]$ , usando a Equação (7). Neste intervalo o  $-1$  indica o menor coeficiente,  $1$  o maior coeficiente e entre esses valores os demais coeficientes [20]. A normalização foi feita em todos coeficientes independente de sua classe.

$$X_{NORM} = (b - a) \cdot \left( \frac{X - \min(X)}{\max(X) - \min(X)} \right) + a \quad (7)$$

onde  $X_{NORM}$  é o valor do coeficiente normalizado,  $X$  é o coeficiente,  $\min(X)$  é o menor coeficiente,  $\max(X)$  é o maior coeficiente,  $a = -1$  e  $b = 1$  são os intervalos de normalização.

Após a classificação dos coeficientes das imagens não houve necessidade de de-normalizar os valores de saída da MLP.

### D. Treinamento e teste da MLP

Como explicado na subseção III-B, as imagens capturadas foram divididas em dois conjuntos, o primeiro, utilizado durante o treinamento da RNA, composto por 90% das imagens e o segundo, utilizado durante a fase de teste, composto pelos os 10% restantes.

Conforme explicado na subseção II-B, o algoritmo de treinamento utilizado foi o de Levenberg-Marquardt. Este algoritmo de treinamento foi executado repetidas vezes com diferentes números de neurônios na camada oculta. A quantidade

inicial de neurônios foi de 3. Esta quantidade foi incrementada em 3 neurônios em cada execução. Para evitar sobre-treino (*overfitting*) foi utilizada a heurística de estabelecer um erro quadrático médio (*MSE*) no valor de  $1 \times 10^{-10}$  e treinar a MLP até que não houvesse mudança significativa no valor do *MSE*. O valor do *MSE* é obtido com a Eq. (8).

$$MSE = \frac{\sum_{i=1}^n (y_i^a - y_i^e)^2}{n} \quad (8)$$

onde  $y^a$  and  $y^e$  são os valores atuais e esperados, respectivamente,  $i$  é a amostra de treino e  $n$  é o número de amostras de treinamento. Nos testes o treinamento da MLP convergiu com 150 épocas. A quantidade de neurônios encontrada foi de 60 neurônios.

#### IV. RESULTADOS

Neste trabalho foram realizados experimentos para se reduzir a quantidade de coeficientes PCA. Foram realizados também, experimentos para se determinar a arquitetura da MLP que mais se adequava à classificação das notas de real.

Cada imagem redimensionada possui uma quantidade de  $300 \times 140$  pixels de largura e altura. Assim, cada imagem possui um total de 42.000 valores. Com a utilização do PCA cada imagem foi reduzida a um total de 500 coeficientes. Isto significa uma redução para  $\approx 1,19\%$  do total de valores.

Os experimentos de teste foram realizados com 50 amostras. A Tabela II ilustra a quantidade de amostras por nota da moeda Real e o percentual de acertos do modelo.

Tabela II. PERCENTUAL DE ACERTOS DO MODELO PARA OS CONJUNTOS DE TESTE

Nota (R\$)	Qtd. Amostras	% Acerto
2	8	100%
5	6	100%
10	5	80%
20	6	100%
50	8	100%
100 v1	9	89%
100 v2	7	86%
<b>Total</b>	<b>50</b>	<b>94%</b>

Observa-se nesta tabela que experimentos com amostras de notas de 2, 5, 20 e 50 reais obtiveram acerto de 100%. Isto demonstra que o modelo identificou satisfatoriamente para amostras destas classes.

Nos experimentos realizados com as notas de 10, 100 v1 e 100 v2 reais obtiveram acertos de 80%, 89% e 86%, respectivamente. Estes percentuais equivalem à quantidade de uma amostra com erro para cada classe. Estes percentuais de acerto para estas classes foram bastante satisfatórias também. Ressalta-se que, as amostras com erro não foram identificadas como nota de outro valor. Desta forma, embora não tenha identificado a nota, evitou-se indicar um valor equivocadamente.

Observa-se ainda que do total de 50 amostras de teste, 47 foram identificadas corretamente e apenas 3 não foram identificadas. O total de acerto geral do modelo foi de 94%.

#### V. CONCLUSÃO

Neste trabalho apresenta-se um modelo para reconhecimento de notas da moeda Real usando PCA e uma RNA do tipo MLP. O sistema recebe imagens digitalizadas de notas de dinheiro e as classifica para identificação de seu valor correspondente. Ressalta-se que o objetivo deste sistema é apenas a identificação do valor das notas e não constatar sua legitimidade.

O PCA se mostrou eficiente na redução e caracterização das imagens de notas Real compactando significativamente o tamanho das imagens. Apesar da redução de dados no processo de obtenção das componentes principais a MLP conseguiu classificar as amostras de notas de forma satisfatória

Os resultados de classificação mostraram-se satisfatórios, com percentuais de acerto de 100% na maioria das classes. Nas classes em que houveram erros, estes percentuais foram muito baixos.

Os resultados encontrados nos experimentos se mostraram bastante promissores. Espera-se utiliza-lo no futuro para auxiliar deficientes visuais.

#### VI. TRABALHOS FUTUROS

Como trabalhos futuros pretende-se:

- Desenvolver uma versão, apenas da etapa de classificação para a plataforma Android;
- adequar a técnica de digitalização para utilização com usuários deficientes visuais;
- experimentar maiores reduções dos coeficientes do PCA e
- experimentar outras técnicas de redução de características e classificação.

#### VII. AGRADECIMENTOS

Este trabalho foi desenvolvido com apoio dos editais 05, 06 e 08 de 2012 da Fundação de Amparo a Pesquisa de Goiás (FAPEG) e do edital 01 de 2011 do Instituto Federal Goiano.

#### REFERENCES

- [1] Aoba, M., Kikushi, T., and Takefuji, Y. (2003). Euro Banknote Recognition System Using a Three-layered perceptron and RBF Networks. *IPJS Transactions on Mathematical Modeling and Its Application*, 44(SIG 7 (TOM 8)).
- [2] Banco Central do Brasil (2013). Museu de Valores do Banco Central do Brasil.
- [3] Chae, S.-H., Kim, J., and Pan, S. (2009). A Study on the Korean Banknote Recognition Using RGB and UV Information. In Ślęzak, D., Kim, T.-h., Chang, A.-C., Vasilakos, T., Li, M., and Sakurai, K., editors, *Communication and Networking*, volume 56 of *Communications in Computer and Information Science*, pages 477–484. Springer Berlin Heidelberg.
- [4] Fukunaga, K. (1990). *Introduction to Statistical Pattern Recognition*. Computer Science and Scientific Computing Series. Academic Press, 2nd edition.
- [5] Hagan, M. and Menhaj, M. (1994). Training feed forward networks with the Marquardt algorithm. In *IEEE Transactions on Neural Networks*, volume 6, pages 861–867. IEEE.

- [6] Hasanuzzaman, F. M., Xiaodong, Y., and YingLi, T. (2011). Robust and effective component-based banknote recognition by SURF features. In *Wireless and Optical Communications Conference (WOCC), 2011 20th Annual*, pages 1–6.
- [7] Haykin, S. (1998). *Neural Networks - A comprehensive Foundation*. Prentice Hall: Upper Saddle River, 2nd edition.
- [8] Hinwood, A., Preston, P., Suaning, G. J., and Lovell, N. H. (2006). Bank note recognition for the vision impaired. *Australasian Physical & Engineering Sciences in Medicine*, 29(2).
- [9] Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *J. Educ. Psychol. Columbia University*, 24:417–441 e 498–520.
- [10] IBGE (2012). Censo 2010. <http://www.censo2010.ibge.gov.br/resultados>.
- [11] Karhunen, K. (1947). *Über lineare Methoden in der Wahrscheinlichkeit-srechnung*. Suomalaisen Tiedeakatemian toimituksia: Mathematica-physica. Universitat Helsinki.
- [12] Levenberg, K. (1944). A method for the solution of certain non-linear problems in least squares. *Quarterly Journal of Applied Mathematics*, II(2):164–168.
- [13] Levy, A. and Lindenbaum, M. (2000). Sequential Karhunen–Loeve Basis Extraction and its Application to Images. *IEEE TRANSACTIONS ON IMAGE PROCESSING*, 9(8):1371–1374.
- [14] Loève, M. (1977). *Probability Theory I*. Number 9780387902104 in Graduate Texts in Mathematics. Springer.
- [15] Marquardt, D. (1963). An algorithm for least squares estimation of non-linear parameters. *Journal of the society of Industrial and Applied Mathematics*, pages 431–441.
- [16] MATLAB (2010). *Version 7.10.0 (R2010a)*. The MathWorks Inc., Natick, Massachusetts.
- [17] Nguyen, D. and Widrow, B. (1990). Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. In *Neural Networks, 1990., 1990 IJCNN International Joint Conference on*, volume 3, pages 21–26. IEEE.
- [18] Sevkli, M., Türkyilmaz, A., and Aksoy, M. S. (2002). Banknote recognition using inductive learning. Proceedings of Int. Conf. On Fuzzy Syst. And Soft Computational Intelligence in Management and Industrial Eng - 15. FSSCIMIE'02, pages 122–128.
- [19] Toyman, I. and Thambidurai, J. (2011). Banknote recognition on Android platform.
- [20] Zanchettin, C. and Ludermir, T. B. (2005). Sistemas neurais híbridos para reconhecimento de padrões em narizes artificiais. *SBA Controle & Automação*, 16(2).