

Protótipo portátil identificador de cédulas de dinheiro

Lucas C. Santos¹, Wendley S. Silva²

¹Engenharia de Computação – Universidade Federal do Ceará (UFC) – Sobral, CE – Brazil

Abstract. *This paper presents the development of a prototype device capable of identifying colors and money in order to facilitate the daily lives of blind or visually impaired people. The prototype consists of a microcontroller and a color sensor that stores the values obtained in a computer and later uses them as the basis of the classification step. The algorithms KNN, SVM, Naive Bayes and Decision Tree were used. The first classification results were satisfactory since no preprocessing of the data was applied. There are still many approaches to be implemented in order to improve the classification success rate as well as techniques to improve sensor reading.*

Resumo. *Neste artigo, apresenta-se o desenvolvimento de um protótipo de um dispositivo capaz de identificar cores e dinheiro com intuito de facilitar o dia a dia de pessoas cegas ou com algum tipo de deficiência visual. O protótipo é composto por um microcontrolador e um sensor de cor que armazena os valores obtidos em um computador e, posteriormente usa-os como base na etapa de classificação. Foram usados os algoritmos KNN, SVM, Naive Bayes e Árvore de Decisão. Os primeiros resultados de classificação foram satisfatórios visto que não foi aplicado nenhum tipo de pré processamento nos dados. Há ainda muitas abordagens a serem implementadas a fim de melhorar a taxa de sucesso na classificação além de técnicas para aperfeiçoar a leitura do sensor.*

1. Introdução

De acordo com os dados do Censo de 2010, no Brasil cerca de 35.774.392 pessoas (18,75% da população nacional) possuem algum tipo de deficiência visual [CENSO 2010], sendo cerca de 506.377 pessoas são incapazes de enxergar (completamente cegos), 6.056.533 de pessoas possuem grande dificuldade permanente de enxergar (baixa visão ou visão subnormal) e outros 29.211.482 declararam possuir alguma dificuldade permanente de enxergar, ainda que usando óculos ou lentes.

Identificação de cores é uma dentre várias dificuldades de um deficiente visual. Outra grande dificuldade, é a identificação de cédulas de dinheiro [Vida Livre 2014]. Embora existam muitos mecanismos para facilitar a diferenciação das cédulas de dinheiro, muitos deficientes visuais relatam a sua dificuldade em identificá-las por estes mecanismos. Há ainda a dificuldade de se ter em circulação as notas da família anterior, as quais têm padrões diferentes de elementos de diferenciação das cédulas atuais.

Assim como o Euro, o Real brasileiro, tanto as novas cédulas quanto a versão anterior das notas, tem diferentes cores e tonalidades de acordo com o seu valor. Este trabalho propõe a criação de um dispositivo identificador de cores e cédulas de dinheiro a partir de um sensor de cor simples e uma etapa de classificação utilizando algoritmos de aprendizagem de máquina, a fim de ajudar deficientes visuais em atividades cotidianas possibilitando-lhes uma maior autonomia, identificando cores de objetos e valores de cédulas de dinheiro, tendo um retorno de forma sonora.

O artigo está organizado como segue. Na Seção 2, descreve-se os trabalhos relacionados a classificação de cédulas e os difere da proposta deste. A Seção 3, mostra os principais passos da construção do protótipo, da aquisição dos dados e construção da base de dados. Na Seção 4, descreve-se toda a etapa de classificação a partir das bases de dados criadas, apresentando a metodologia e os resultados para cada algoritmo de classificação abordado. Por fim, a Seção 5 conclui o trabalho e sugere ajustes futuros na metodologia.

2. Trabalhos relacionados

Existem outras contribuições com abordagens semelhantes que buscam solucionar o problema de classificar cédulas de dinheiro a fim de ajudar na autonomia de pessoas cegas. Em [Sousa et al. 2018] é abordado um estudo, com processamento digital de imagens, de cédulas de diferentes países, como o Real brasileiro, Dólar americano e Euro, usando como base de dados imagens capturadas das cédulas de dinheiro e classificando-as segundo suas respectivas cores e outros elementos como textos e fragmentos de imagens.

Em [Mombach and Welfer 2017] foi construído um aplicativo para plataforma móvel que captura a imagem da cédula de dinheiro usando o sensor da câmera do aparelho celular e retorna o resultado da classificação de forma sonora pelo próprio *smartphone*. O trabalho também aborda muito bem o processo iterativo de testes com voluntários cegos, que ajudaram no aprimoramento do aplicativo.

Em ambos trabalhos citados, a abordagem inicia obtendo a imagem da cédula de dinheiro a partir de uma câmera fotográfica, por exemplo a câmera do *smartphone*, sendo extraídos muitos atributos além da cor média, como letras e números presentes em fragmentos de imagem. Por se tratar de um aplicativo móvel, há uma certa dificuldade nos quesitos preço e usabilidade do *smartphone* apresentada no próprio artigo. Já este trabalho, busca a implementação de um dispositivo portátil, de simples manuseio e de baixo custo, utilizando apenas a cor (R, G, B) como característica extraída da cédula de dinheiro a partir de um simples sensor de cor compatível com Arduino.

3. Desenvolvimento

O dispositivo tem o intuito de capturar a cor do objeto posicionado à sua frente e guardá-los em uma base para, posteriormente, serem analisados e classificados por algoritmos de inteligência artificial.

Posteriormente é desejada a miniaturização do protótipo e que todo o processamento esteja apenas dentro do dispositivo, já que atualmente o processamento de captura está no microcontrolador e o de classificação está no computador.

3.1. Materiais utilizados

1. **Arduino UNO:** Foi escolhida esta plataforma como microcontrolador do protótipo devido uma maior familiaridade dos autores com a ferramenta, além de facilitar na implementação física por conter muitas portas digitais e analógicas e dispor de uma interface USB para comunicação e alimentação, também na implementação lógica por conter muitas funções já implementadas internamente.
2. **Sensor de cor TCS3200:** Por ser bem acessível e ter um funcionamento de fácil compreensão, este foi o sensor de cor escolhido para o projeto. Apesar de, a princípio, ter sido pensado na construção de um sensor de cor mais simples de

forma caseira com apenas um LED RGB e um LDR, esta ideia ainda não foi terminada e foi decidido continuar com o projeto, a priori, com o sensor de cor TCS3200.

3. **Fios e conectores:** Por fim, foram utilizados fios para conexão de alimentação e comunicação entre o Arduino e o sensor.

3.2. Construção do protótipo

Com o intuito de manter o sensor em uma posição e luminosidade fixa/controlada para que todas as leituras sejam obtidas sob as mesmas condições, foi elaborada uma caixa em madeira, onde se encontram o sensor de cor e o microcontrolador, ambos fixados e conectados entre si. Como mostrado na Figura 1, o sensor encontra-se na parte superior e o microcontrolador na parte interna da caixa, tendo apenas sua interface USB à mostra, onde conecta-se o cabo para alimentação e comunicação serial do dispositivo com o computador. O protótipo tem seguintes dimensões, 46mm x 70mm x 83mm. Pretendendo ser otimizado, diminuindo seu tamanho e dispor de uma bateria interna para se tornar portátil futuramente.



Figura 1. Protótipo criado

3.3. Calibragem do sensor

Concluída a etapa de construção do protótipo, foi necessária a calibragem do sensor para corrigir, via código, alguns fatores, como por exemplo, a distância do sensor para o objeto que está sendo lido.

Esta calibragem foi empírica e no código do microcontrolador, ajustando os valores dos limiares para conversão da saída do sensor para valores de cada canal de cor do objeto (R, G, B).

De acordo com o *datasheet* [TAOS 2009], o sensor é composto por 64 fotodiodos, sendo que 16 tem filtros da cor vermelha, 16 para a cor verde, 16 para a cor azul e 16 não contém filtro, que são distribuídos uniformemente pelo sensor e são utilizados para captar a intensidade luminosa de cada canal de cor refletida pelo objeto a sua frente. No módulo que contém o sensor de cor, também contém um conjunto de 4 LEDs brancos de alto brilho para iluminação do objeto. O controle dos filtros e da saída do sensor é feito por 5 pinos (S1, S2, S3, S4 e OUT), sendo os dois primeiros para configuração da escala de frequência de saída e os outros dois para designar qual canal será lido. A saída do sensor é obtida pelo pino (OUT).

3.4. Comunicação com o computador

Como mencionado anteriormente, o protótipo é conectado a um cabo USB que é usado como sua alimentação e meio de comunicação para o computador. Sendo assim, todos os valores obtidos e convertidos do sensor são mandados pela serial do Arduino e recebidos por um *script* escrito em Python executando no computador, para isto, foi usada a biblioteca PySerial [Liechti 2015].

Nos testes iniciais, os *scripts* feitos para "escutar" a comunicação serial, ao receber os valores da cor lida no sensor, mostra a cor colorindo a tela do computador para comparação da cor com o objeto real, a fim de analisar a precisão e fidelidade do sensor e da conversão, e assim realizar ajustes caso haja necessidade.

3.5. Construção da base de dados

Para a construção da base de dados, foi criado outro *script* em Python, baseado no primeiro, para que os dados enviados pelo microcontrolador sejam armazenados automaticamente em um arquivo CSV, para posteriormente ser usado como base principal dos algoritmos de classificação no projeto.

O arquivo CSV (*Comma Separated Values*) é composto por 5 colunas, cada uma representando um atributo das amostras da base, sendo eles:

- ID:** Usado apenas como identificador único e contador das amostras da base.
- R:** Valor referente à intensidade do canal Vermelho da amostra lida pelo sensor.
- G:** Valor referente à intensidade do canal Verde da amostra lida pelo sensor.
- B:** Valor referente à intensidade do canal Azul da amostra lida pelo sensor.
- Classe:** Este sendo o único atributo da base que é requisitado para ser passado manualmente enquanto recebe os valores lidos do sensor. Refere-se em que grupo aquela amostra será incluída. Diferentemente dos atributos anteriores, ditos como previsores, este atributo é o objetivo que se quer alcançar na classificação. Este atributo é necessário devido aos algoritmos utilizados neste projeto serem de aprendizagem supervisionada.

4. Classificação

Todos os algoritmos e estratégias de aprendizagem de máquina para classificação foram implementados em Python utilizando as rotinas da biblioteca scikit-learn [Pedregosa et al. 2011].

Para todos os algoritmos de classificação, foram utilizadas a estratégia K-Fold para validação cruzada com $K = 10$ sem pré-processamento nos dados, foram utilizados os próprios valores lidos e convertidos do sensor e, selecionados apenas os atributos **R**, **G** e **B** da base como atributos previsores e o atributo **Classe** como atributo objetivo a ser previsto pelos modelos de classificação utilizados.

Foram usados os principais algoritmos de classificação conhecidos, para realização dos testes neste trabalho, como por exemplo KNN, SVM, Árvore de Decisão, Naive Bayes.

Como resultado, os *scripts* criados retornam a acurácia de cada iteração do K-Fold e a acurácia média de cada algoritmo, com o intervalo de confiança de 95%.

4.1. Base de cores sólidas

Neste primeiro teste, foi utilizada uma base criada a partir da leitura do sensor de 10 cartões coloridos com cores sólidas (sem transição de uma cor para outra) diferentes umas das outras, sendo realizadas 10 leituras para cada cartão, originando uma base com 100 amostras. As cores dos cartões usados foram *vermelho*, *vermelho escuro*, *verde*, *verde escuro*, *azul*, *azul escuro*, *cinza*, *branco*, *preto* e *amarelo* como pode ser visto no gráfico de dispersão na Figura 2.

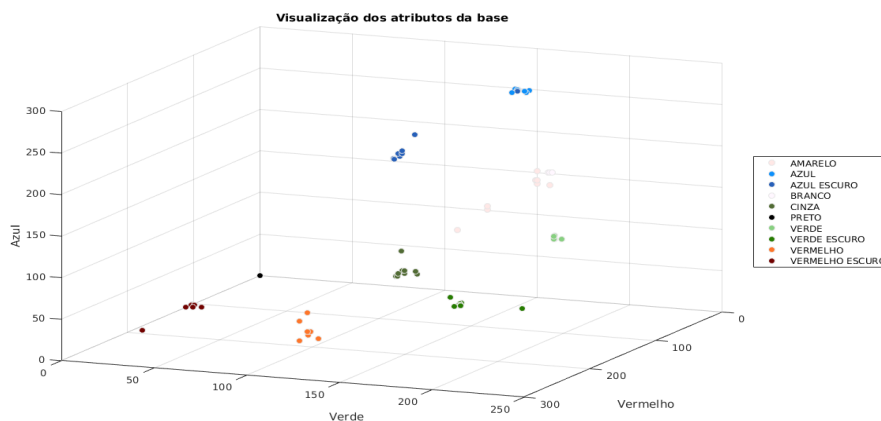


Figura 2. Gráfico de dispersão da base de cores sólidas

4.1.1. Resultados

Como visto na Figura 2, para esta base as amostras ficaram bastante dispersas entre classes e próximas umas das outras de mesma classe, o que resultou em uma boa classificação por todos os algoritmos testados.

As configurações dos algoritmos de classificação que resultaram, em média, nas maiores taxas de acerto foram

- **KNN**: Usando $K = 5$ vizinhos mais próximos por distância euclidiana
- **SVM**: Usando kernel linear e constante de penalidade igual a 1
- **Árvore de Decisão**: Usando critério da entropia do máximo ganho
- **Naive Bayes**: Naive Bayes Gaussiano

Sendo que o algoritmo KNN obteve a maior acurácia média dentre os quatro classificadores usados, como é mostrado na Tabela 1. A Figura 3 mostra os resultados a cada iteração do K-Fold.

	Médias	Desvio Padrão
KNN	97 %	0.0458
SVM	96 %	0.0490
Árvore de Decisão	96 %	0.0490
Naive Bayes	93 %	0.0640

Tabela 1. Acurácias médias de cada classificador na base de cores sólidas

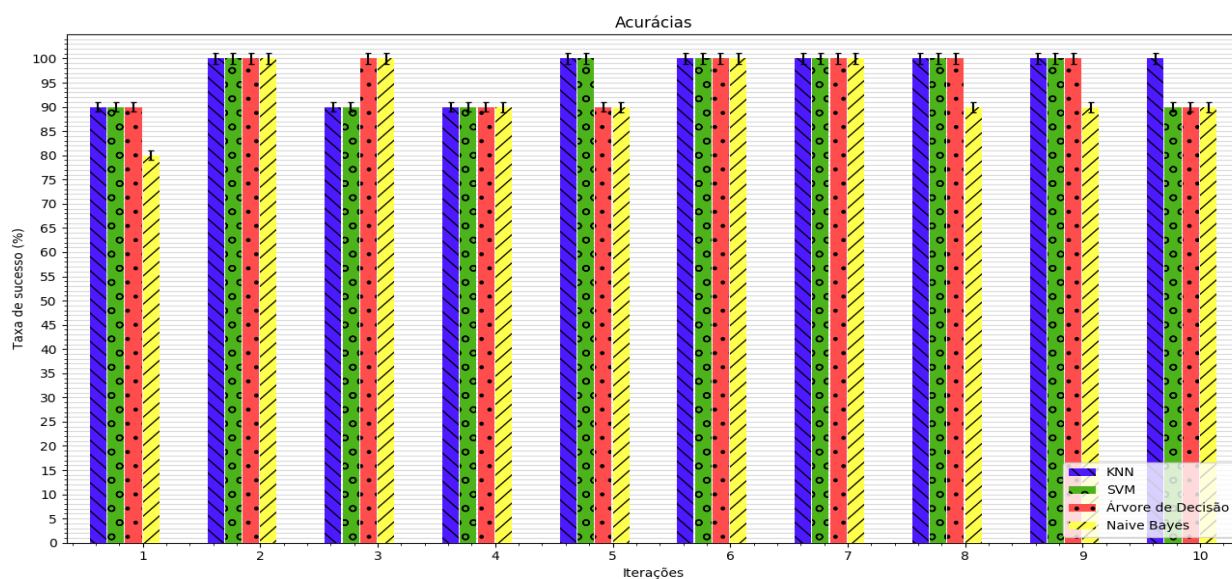


Figura 3. Resultados dos classificadores na primeira base

4.2. Base de cédulas de dinheiro

Ao término do primeiro teste, foi colocado em prática a proposta inicial do projeto, a tentativa de classificar cédulas de dinheiro, mais especificamente do real brasileiro, por sua cor.

Igualmente na etapa anterior, a base foi criada sendo realizada a leitura dos dois lados da cédula, passando a nota de uma extremidade a outra a fim de pegar todas as diferentes tonalidades presentes na cédula e, assim criar uma base mais abrangente.

A base foi criada com os 4 menores valores de cédulas disponíveis atualmente, 2 reais, 5 reais, 10 reais e 20 reais. Sendo realizadas 10 leituras em cada cédula, originando uma base com 40 amostras, como é mostrado no gráfico de dispersão na Figura 4.

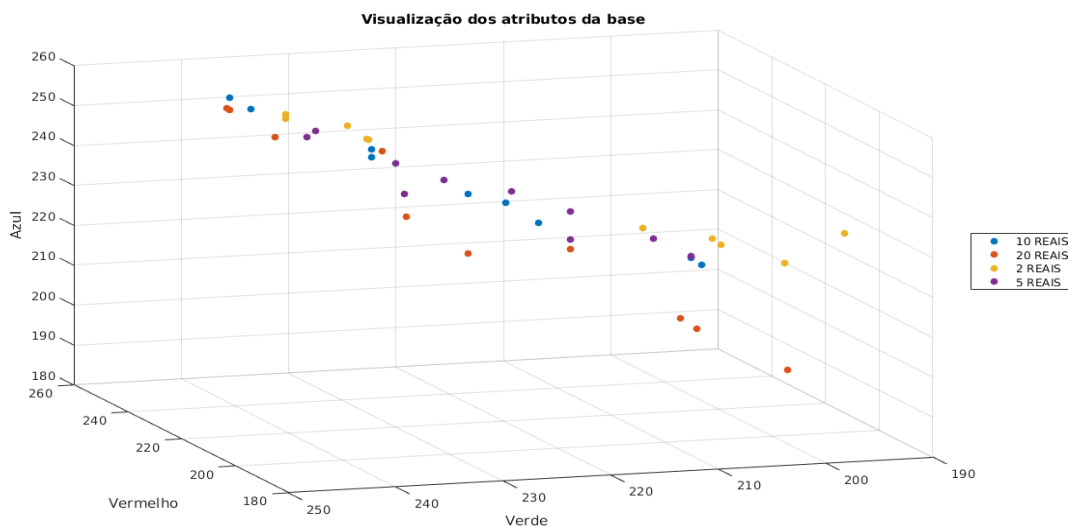


Figura 4. Gráfico de dispersão da base de cores das cédulas

4.2.1. Resultados

Como visto na Figura 4, esta segunda base teve bem menos amostras e ficaram muito próximas das outras classes da base e não tão próximas assim umas das outras de mesma classe, se comparado a primeira base analisada, cenário este que dificultou muito a taxa de acertos dos classificadores.

As configurações dos algoritmos de classificação que resultaram, em média, nas maiores taxas de acerto para esta base foram

- **KNN**: Usando o primeiro vizinho mais próximo por distância euclidiana
- **SVM**: Usando kernel linear e constante de penalidade igual a 1
- **Árvore de Decisão**: Usando critério da entropia do máximo ganho
- **Naive Bayes**: Naive Bayes Gaussiano

O algoritmo SVM obteve a maior acurácia dentre os quatro classificadores usados, como é mostrado na Tabela 2. A Figura 5 mostra os resultados a cada iteração do K-Fold.

	Médias	Desvio Padrão
KNN	67.50 %	0.2350
SVM	82.50 %	0.1601
Árvore de Decisão	62.50 %	0.2250
Naive Bayes	42.50 %	0.2750

Tabela 2. Acurácias médias de cada classificador na base de cédulas de real

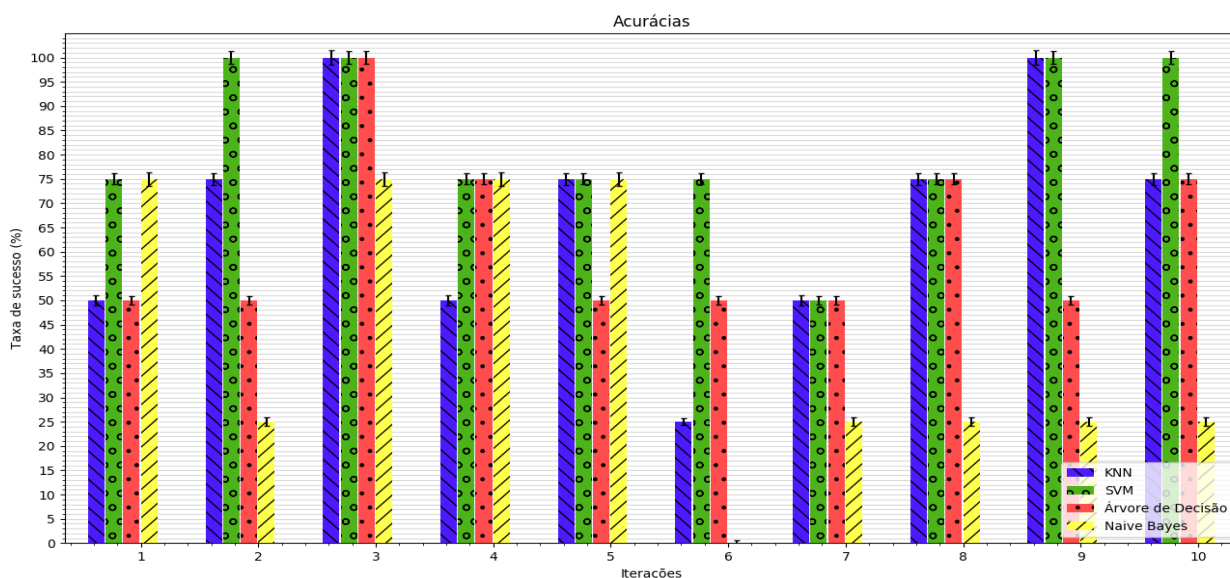


Figura 5. Resultados dos classificadores na segunda base

5. Conclusão

Pode-se notar um grande desafio na classificação de cédulas por sua cor a partir de um sensor de cor simples, diferentemente da classificação de cores sólidas obtidas pelo mesmo sensor. O fato de não haver nenhum tipo de pré-processamento nos dados antes da aprendizagem, como a extração de novos atributos/características da base, ou mesmo pelas

próprias características das cédulas de dinheiro que têm uma grande transição de cores, embora sejam da mesma tonalidade (ou próximas), o que pôde acarretar em pontos não tão espaçados entre classes no espaço, dificultando ainda mais o sucesso na classificação. Pretende-se elaborar novos ajustes e testes tanto na criação da base utilizando uma melhor conversão, tanto utilizando um pré-processamento na base antes da aprendizagem. O intuito final do projeto é a criação de um protótipo autônomo capaz de realizar a captura e classificação instantânea dos valores obtidos e retornar ao usuário em forma de áudio, o valor da cédula lida.

Referências

- Arduino e Cia (2014). Sensor de reconhecimento de cor TCS230 / TCS3200. <https://www.arduinoecia.com.br/2014/02/sensor-de-reconhecimento-de-cor-tcs230.html>. Acesso em: 04/07/2019.
- CENSO (2010). Censo 2010. <https://www.ibge.gov.br/estatisticas-novoportal/sociais/populacao/2098-np-censo-demografico/9662-censo-demografico-2010.html?edicao=9749&t=destaques>. Acesso em: 11/11/2018.
- Liechti, C. (2015). pySerial's documentation. <https://pythonhosted.org/pyserial/>. Acesso em: 01/06/2019.
- Mombach, J. and Welfer, D. (2017). Aplicativo identificador de cédulas para apoio ao ensino do sistema monetário brasileiro a pessoas com deficiência visual. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 28, page 1067.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- Sousa, L. P. d., Veras, R. d. M. S., Vogado, L. H. S., and Neto, L. d. S. B. (2018). Metodologia de identificação de cédulas monetárias para deficientes visuais. *Revista de Sistemas e Computação-RSC*, 8(1).
- TAOS (2009). TCS3200, TCS3210 PROGRAMMABLE COLOR LIGHT-TO-FREQUENCY CONVERTER. <https://www.mouser.com/catalog/specsheets/TCS3200-E11.pdf>. Acesso em: 03/07/2019.
- Vida Livre (2014). Novas Cédulas de Real não ajudam os deficientes visuais. <https://vidamaislivre.com.br/2014/01/14/novas-credulas-de-real-nao-ajudam-os-deficientes-visuais-%C2%9D/>. Acesso em: 11/11/2018.