

Desenvolvimento de um esquema de recepção para sistemas *Optical Camera Communications* (OCC) usando dispositivos *Android*

Matheus F. Alves, Leandro R. Ximenes e Igor S. C. Rodrigues

Resumo— Este trabalho desenvolve e implementa um inédito esquema de recepção *Open Source* para sistemas de comunicação baseado na tecnologia de Comunicações por Câmeras Ópticas (OCC - *Optical Camera Communications*) do tipo Tela-para-Câmera (S2C - *Screen-to-Camera*). Foi desenvolvido um aplicativo *Android* para câmeras utilizando a plataforma *Android Studio* que captura imagens contendo *bits* codificados nos seus *pixels*, exibidas por uma tela composta por Diodos Emissores de Luz (LEDs - *Light Emitting Diodes*), e um algoritmo de processamento e recuperação de imagens implementadas no MATLAB que decodifica os dados enviados.

Palavras-Chave— Comunicações por Câmeras Ópticas, Tela-para-Câmera, Processamento Digital de Imagens.

Abstract— This work develops and implements an unprecedented *Open Source* reception scheme for a communication system based on *Optical Camera Communication* (OCC) technology of the *Screen-to-Camera* (S2C) type. An *Android* camera application captures S2C-encoded video sequences exhibited by a *Light Emitting Diode* (LED) screen, and image processing algorithms in the MATLAB environment restore and recover the transmitted data.

Keywords— *Optical Camera Communications*, *Screen-to-Camera*, *Digital Image Processing*.

I. INTRODUÇÃO

A evolução das comunicações móveis criou a necessidade pela busca de alternativas que pudessem ajudar a lidar com a quantidade massiva de dados gerados a um baixo custo de implementação/operação e alta eficiência energética [1].

Nesse contexto, o uso da tecnologia OCC é oportuno. No OCC, um arranjo de LEDs é utilizado como transmissor emitindo sinais luminosos através do espaço livre (ar) e uma câmera, presente na maioria dos *smartphones* atuais, como receptor [1]. A Fig. 1 mostra um diagrama de blocos generalizado de um sistema OCC.

Uma subcategoria do OCC é denominada OCC-S2C, onde uma tela formada por LEDs envia dados a partir da exibição de seqüências de quadros contendo *bits* codificados em cada um dos seus *pixels*. Dentre suas aplicações, destacam-se a comunicação veicular, sinalização digital e localização *indoor* [1]. No OCC-S2C, o transmissor adquire dualidade de funções, realizando comunicação enquanto exibe imagens [2].

Este projeto desenvolve e implementa um esquema de recepção para sistemas baseados na tecnologia OCC-S2C. Foi desenvolvida uma interface baseada no sistema operacional *Android* que realiza a captura de vídeos para processamento e decodificação dos símbolos (conjunto de *bits*) codificados nos seus quadros.

Além disso, um algoritmo de restauração de imagens, com detecção manual de Região de Interesse (RoI - *Region of Interest*) e capacidade de sincronização baseado na modulação *On-Off Keying* (OOK), foi desenvolvido no MATLAB para tratar os quadros de um vídeo codificado segundo uma abordagem tensorial [3]. Este procedimento foi inserido antes da etapa de decodificação, realizada utilizando o algoritmo de Mínimos Quadrados Alternados (ALS - *Alternating Least Squares*) [4].

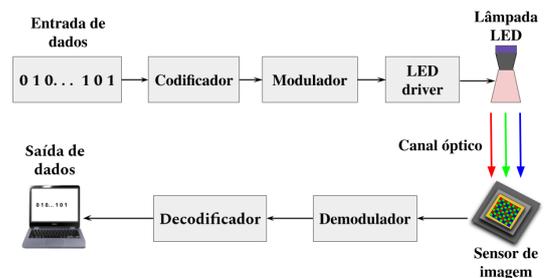


Fig. 1. Cadeia de um típico sistema OCC.

II. METODOLOGIA

A Fig. 2 exibe um diagrama de blocos que descreve as etapas de desenvolvimento do esquema de recepção proposto.

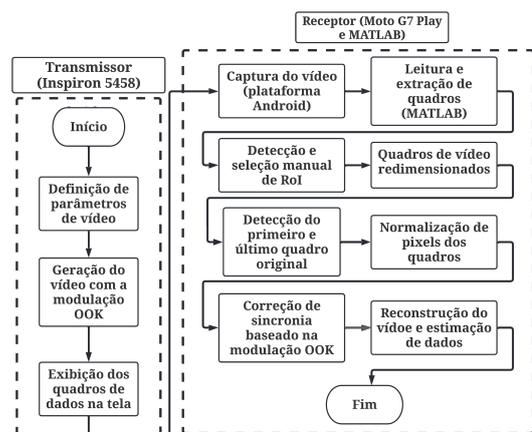


Fig. 2. Diagrama de blocos do *setup* experimental.

Inicialmente, foi criado o aplicativo *Android* instalado no *smartphone* que grava vídeos e permite a modificação de alguns parâmetros da câmera como a taxa de quadros, resolução do vídeo, a taxa de captura e o ajuste do foco.

Depois, um algoritmo que processa o vídeo capturado pelo aplicativo e reconstrói-o para que se torne o mais próximo possível do vídeo original foi desenvolvido no MATLAB.

A Fig. 3 mostra o aplicativo aberto no *smartphone*, no *setup* experimental de validação da plataforma. A Fig. 4 exhibe exemplos de um quadro processado a cada etapa do algoritmo, desde sua geração até a sua reconstrução na saída do receptor.

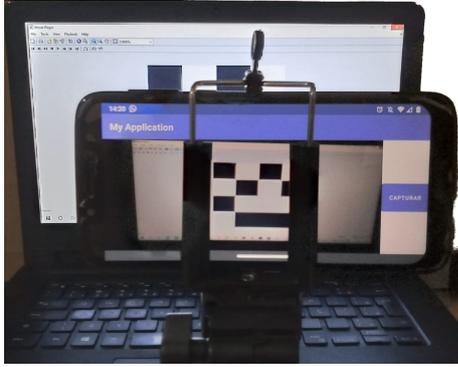


Fig. 3. Visualização do *app* e esquema proposto.

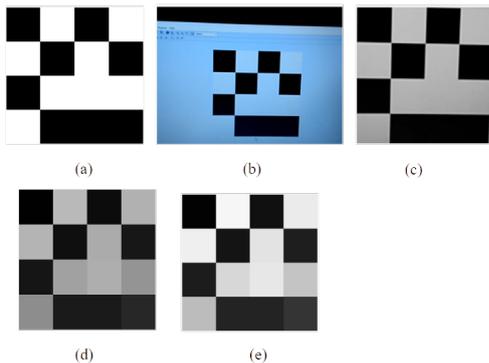


Fig. 4. (a) - Quadro original; (b) - Quadro capturado pelo aplicativo da câmera; (c) - Identificação manual da RoI em (b); (d) - Quadro redimensionado; (e) - Quadro normalizado.

III. RESULTADOS E DISCUSSÕES

Usando o esquema proposto, foi possível obter curvas de Erro Quadrático Médio Normalizado (*NMSE* - *Normalized Mean Square Error*) dos símbolos recebidos.

As curvas usadas para validação foram geradas em função de um parâmetro que controla a Relação Sinal-Ruído (SNR - *Signal-to-Noise Ratio*) de recepção. O ruído branco Gaussiano para o cálculo do SNR foi gerado artificialmente na entrada do receptor, embora ruídos inerentes do experimento prático certamente estavam presentes.

Para cada valor de SNR testado, um conjunto de capturas de vídeo foram realizadas alterando-se o nível de luminosidade do ambiente de testes, um quarto residencial. O nível de luminosidade é medido em *lux* utilizando o aplicativo Luxímetro, disponível na *Google Play Store* [5].

A análise foi realizada contemplando dois casos. Um em que é desprezível a interferência de luz ambiente na comunicação (i.e., quase totalmente escuro, com 30 *lux*) e outro em que

outras fontes de luz artificiais (e.g., lâmpadas convencionais) estão presentes no momento da captura dos quadros (i.e., 300 *lux*). O desempenho de recepção dos *bits* foi verificado para o caso com e sem uma etapa de sincronia desenvolvida.

Um dos resultados é mostrado na Fig. 5. Fica evidente que, sem o esquema de sincronia proposto, a recuperação e restauração dos símbolos originais ficam fortemente prejudicadas, uma vez que as curvas tracejadas (sem sincronia) caem apenas lentamente com o SNR (dB) em comparação com as curvas sólidas (com sincronia). Tal afirmação é válida para ambos os níveis de luminosidade analisados. Também, é possível notar que maiores níveis de interferência luminosa aumentam a potência do ruído, piorando a estimação dos símbolos.

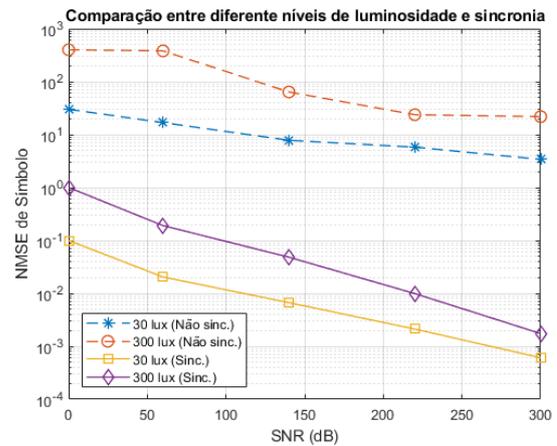


Fig. 5. NMSE de símbolo (em dB) com e sem sincronia em diferentes níveis de iluminação.

IV. CONCLUSÕES

Este projeto desenvolveu e implementou uma plataforma funcional de recepção em sistemas OCC-S2C. Os resultados demonstram que esta plataforma *Open Source* é funcional.

Pode-se adaptar a plataforma para diferentes aplicações, tais como a comunicação veicular, implementando técnicas de visão computacional para detecção e seleção de RoI automática e com o uso de diferentes tipos de modulação e codificação buscando incrementar a taxa de dados e robustez do sistema.

REFERÊNCIAS

- [1] T. Nguyen, A. Islam, T. Hossain and Y. M. Jang, "Current Status and Performance Analysis of Optical Camera Communication Technologies for 5G Networks," in *IEEE Access*, vol. 5, pp. 4574-4594, 2017, doi: 10.1109/ACCESS.2017.2681110.
- [2] Tianxing Li, Chuankai An, Xinran Xiao, Andrew T. Campbell, and Xia Zhou. 2015. Real-Time Screen-Camera Communication Behind Any Scene. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys '15)*. Association for Computing Machinery, New York, NY, USA, 197–211. <https://doi.org/10.1145/2742647.2742667>
- [3] L. Ximenes and B. de A. Larêdo, "A Tensor-Based Optical Camera Communication (OCC) System With Joint Data Detection and Video Restoration". *TechRxiv*, 10-Sep-2021, doi: 10.36227/techrxiv.16586357.v1.
- [4] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis and C. Faloutsos, "Tensor Decomposition for Signal Processing and Machine Learning," in *IEEE Transactions on Signal Processing*, vol. 65, no. 13, pp. 3551-3582, 1 July 2017, doi: 10.1109/TSP.2017.2690524.
- [5] *Crunch Bytebox, Light Meter*. Disponível em: crunchybytebox.de/app.php?id=lightmeter