

FERRAMENTA PARA MANIPULAÇÃO E ANÁLISE DE LIGHT FIELDS

GIORDANO ESTÊVÃO BORGES LEAL; RUHAN CONCEIÇÃO; GUILHERME
CORREA; BRUNO ZATT; LUCIANO AGOSTINI; MARCELO PORTO

Universidade Federal de Pelotas – Video Technology Research Group (ViTech)
{gbleal, radconceicao, gcorrea, zatt, agostini, porto}@inf.ufpel.edu.br

1. INTRODUÇÃO

Smartphones estão difundidos globalmente na atualidade. Dispositivos eletrônicos são facilmente encontrados no mercado, vendidos para pessoas de todas as idades, desenvolvendo uma cultura centrada em mídia digital. Com a população global estando cada vez mais imersa em tecnologias cibernéticas as grandes companhias têm estabelecido uma busca por tecnologias imersivas. Realidade virtual – que se foca na imersão total do usuário – e realidade aumentada – que se foca na imersão parcial do usuário – como exemplos dessas, têm recebido suporte nos últimos anos.

Além dessas tecnologias televisões 3D têm oferecido uma experiência visual distinta para o usuário, oferecendo uma nova dimensão de percepção. Enquanto sua premissa atraiu investimentos de grandes corporações, a recepção do público geral durante os últimos anos tem sido abaixo do esperado – o que reflete a natureza falha da utilização de métodos óticos não visualmente “naturais” em representações visuais. A tecnologia de vídeos 3D hoje em dia está em um impasse, considerando quanta fadiga ocular elas causam é gerada uma necessidade no mercado de uma melhoria considerável.

Uma tecnologia que propõe um aprimoramento para tais representações é *light fields* (LEVOY, 2006). Essa é originária da função plenótica (ADELSON; BERGEN, 1991) (que visa registrar todo o universo observável) com valores discretizados. Os *light fields* expressam uma imagem e seus muitos detalhes angulares, capturando importante informação óptica. Esses podem ser aplicados em realidade virtual e realidade aumentada, fornecendo mais dados úteis em seu desenvolvimento e diminuindo o *Vergence-Accommodation-Conflict* (VAC) (HOFFMAN; BANKS, 2010) – que é o maior fator que influencia no desconforto visual criado pela visualização de mídia tridimensional – por meio de uma transição angular mais suavizada.

Tendo isso em consideração, *light fields* são uma tecnologia nova importante de ser desenvolvida. Enquanto é possível se achar diversos repositórios com *light fields* online, disponíveis ao público faltam ferramentas que ajudam a manusear eles. Assim, nesse trabalho foi desenvolvida uma série de funções úteis para esse fim, utilizando diversas técnicas para a manipulação dos dados gerados pelos *light fields* nos vários repositórios disponíveis.

2. METODOLOGIA

O código foi escrito utilizando programação orientada a objetos no programa MatLab, no qual cada classe foi desenvolvida considerando os elementos mais utilizados na pesquisa com *light fields*, com o apoio da *Toolbox* “Light Field Toolbox v0.4” (DANSEREAU, 2015). As classes criadas podem ser divididas em quatro grupos: classes base (CB); classes de análise residual (CAR); classes de manipulação de vistas (CMV); e classes de conversão de arquivos YUV (CYUV).

CBs são classes de armazenamento de *light fields* como objetos e manipulações básicas desses. CARs são classes que calculam os resíduos entre as vistas de um *light field*. CMVs são classes que projetam as vistas de um dado

light field em vários planos. E CYUVs são classes que convertem *light fields* para o formato YUV - com ajuda da *Toolbox* “YUV files reading and converting” (SPRLJAN, 2012) - utilizado em codificação de vídeo.

A estrutura das classes permite flexibilidade de criação de métodos específicos para qualquer aplicação específica utilizando *light fields*. A seguir serão mostrados modelos UML das classes, cada uma com nome, seguido de seus atributos, seguido de seus métodos. Os atributos e métodos privados estão sinalizados por um sinal de menos, os públicos com um sinal de mais e os protegidos (*protected*) com um sinal de *hashtag*.

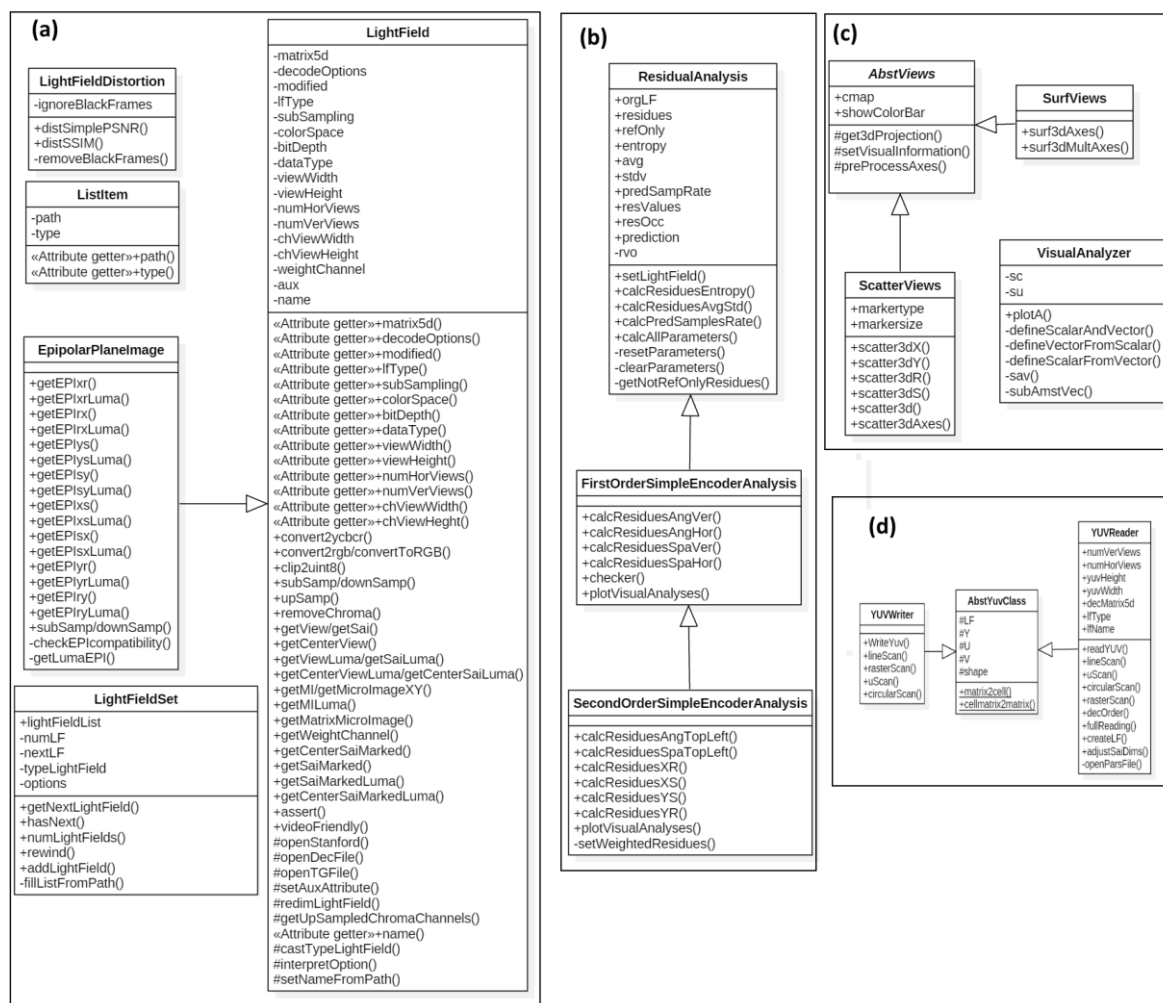


Figura 1. Modelo UML para classes base (a), classes de análise residual (b), classes de manipulação de vista (c) e classes de conversão de arquivos YUV (d).

Na Figura 1 (a) podemos ver as CBs, com a maior delas sendo a classe LightField, que constrói e armazena o *light field* do *dataset* desejado. Herdeira dessa é a CB EpipolarPlaneImage, que coloca o *light field* para o plano epipolar (BOLLES; BAKER; MARIMONT, 1987). A CB LightFieldSet cria uma lista de objetos LightField para fácil acesso de um conjunto numeroso de *light fields* utilizando objetos da CB ListItem. Por fim a CB LightFieldDistortion calcula os valores de PSNR e SSIM entre dois dados objetos LightField, afim de calcular as diferenças entre eles.

Temos na Figura 1 (b) as CARs, com a classe principal do tipo `ResidualAnalysis`, que é herdada por `FirstOrderSimpleEncoderAnalysis`, que, por sua vez, é herdada por `SecondOrderSimpleEncoderAnalysis`. `ResidualAnalysis` é a classe principal, que prepara os métodos necessários para cálculo de resíduos. `FirstOrderSimpleEncoderAnalysis` calcula os resíduos em uma direcção (horizontal

ou vertical) e SecondOrderSimpleEncoderAnalysis calcula os resíduos em duas direções.

Pode se ver a estrutura das CMVs na Figura 1 (c), no qual existe a classe abstrata AbstViews, quer é herdada por SurfViews (que projeta as vistas num plano tridimensional) e ScatterViews (que projeta as vistas num plano bidimensional). Por fim, a classe VisualAnalyzer constrói um objeto ScatterViews e outro SurfViews e os manipula. Por fim na Figura 1 (d) temos as CYUVs, que tanto a classes de escrita (YUVWriter) quanto a classe de leitura (YUVReader) são herdeiras da classe abstrata AbstYuvClass.

3. RESULTADOS E DISCUSSÃO

Podemos ver na Tabela 1 os resultados para um conjunto de *light fields* da análise residual de primeira ordem. São mostrados os valores de entropia que quanto mais perto de 1 maior a similaridade entre todas as vistas de um dado *light field*, média dos resíduos e desvio padrão. Quanto menor a média e desvio padrão, maior a similaridade entre vistas. Quanto maior a semelhança entre as vistas, mais pode ser explorada a codificação em um *light field*.

Nota-se que a entropia, média e desvio padrão dos diversos *light fields* varia consideravelmente. Red_& White_Building é o *light field* com melhor entropia e menor desvio padrão angular vertical, por ser um *light field* com pouca diferença angular vertical entre suas vistas, porém a menor média é pertencente a Pillars – que apresenta maior homogeneidade vertical.

O *light field* com melhor entropia angular horizontal é Danger_de_Mort, que tem padrões repetidos horizontalmente. A menor média angular horizontal foi de Spear_Fence_1 – com várias áreas homogêneas em suas vistas – e o menor desvio padrão foi de Rolex_Learning_Center, com menor diferença entre seus ângulos horizontais.

Tabela 1. Resultados da análise de primeira ordem de dados *light fields*.

Nome do Light Field	Angular Vertical			Angular Horizontal		
	Entropia	Média	Desvio	Entropia	Média	Desvio
-						
Pillars	0,91203	-0,11303	4,7086	0,87908	-0,11352	3,3352
Red_& White_Building	0,99858	-0,25984	3,8173	0,96506	0,19201	3,5105
Rolex_Learning_Center	0,9928	-0,14404	5,1026	0,90025	0,060731	3,2336
Danger_de_Mort	1,0104	-0,3753	4,4468	1,0073	-0,04446	3,8889
Perforated_Metal_1	1,2563	-0,76377	7,2224	1,1857	-0,03751	5,3093
Spear_Fence_1	1,0136	-0,31222	6,0412	0,97082	0,000869	5,6104

Na figura 2 podemos ver um resultado do método plotA de ScatterViews para o *light field* “Desktop” - pertencente à *dataset* EPFL, no qual uma vista é cortada horizontalmente – Figura 2 (a) - e analisada – Figura 2 (b) e (c). Na Figura 2 (b) é possível perceber uma projeção bidimensional em que existem várias cores para esse dado corte, com várias áreas similares. Na projeção tridimensional - Figura 2 (c) - é possível ver nos diferentes ângulos como a imagem se comporta, tendo similaridades entre eles em algumas regiões específicas com maior homogeneidade, e também as diferenças entre transições.

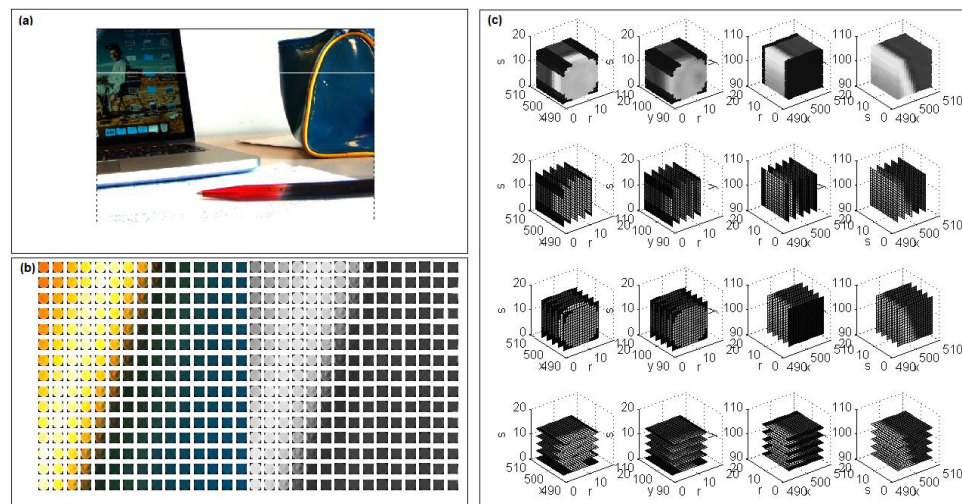


Figura 2. Um dos resultados do método “plotA” na classe VisualAnalyzer.

4. CONCLUSÕES

Light fields é uma tecnologia recente que permite descrever uma imagem e seus diversos ângulos com precisão, capturando informação visual preciosa em uma cena. Para manipular *light fields* são necessárias ferramentas específicas para seu uso.

Difícilmente são encontrados recursos para a pesquisa de *light fields*. Com o intuito de melhorar essa realidade foram criadas funções no programa MatLab que facilitam o trabalho com eles.

Foram então criadas classes base – onde são armazenados os *light fields*, classes de análise residual – que calculam a diferença entre vistas de um *light field*, classes de manipulação de vistas – que permitem acessar as vistas de diversas formas, explorando as características dos *light fields*, e classes de conversão de arquivos YUV – que facilitam a pesquisa utilizando codificadores de vídeo. Como trabalhos futuros propõe-se criar uma interface para a utilização das classes e expandi-las.

5. REFERÊNCIAS BIBLIOGRÁFICAS

- ADELSON, E. H.; BERGEN, J. R. The plenoptic function and the elements of early vision. **Computational Models of Visual Processing**, p. 3–20, 1991.
- BOLLES, R. C.; BAKER, H. H.; MARIMONT, D. H. Epipolar-plane image analysis: An approach to determining structure from motion. **International Journal of Computer Vision**, v. 1, n. 1, p. 7–55, 1987.
- DANSEREAU, D. **Light Field Toolbox v0.4** MathWorks, 2015. Disponível em: <<http://www.mathworks.com/matlabcentral/fileexchange/49683>>
- HOFFMAN, D. M.; BANKS, M. S. Vergence–accommodation conflicts hinder visual performance and cause visual fatigue. v. 8, n. 3, p. 1–48, 2010.
- LEVOY, M. Light fields and computational imaging. **Computer**, v. 39, n. 8, p. 46–55, 2006.
- SPRLJAN, N. **YUV files reading and converting - File Exchange - MATLAB Central**.