



**UNIVERSIDADE DO VALE DO RIO DOS SINOS**

Programa de Pós-Graduação em Geologia

Área de Concentração em Geologia Sedimentar

Linha de Pesquisa: Sensoriamento Remoto e Modelagem Geológica

## Dissertação de Mestrado

# ***OUTCROP EXPLORER: UM SISTEMA DE RENDERIZAÇÃO BASEADA EM PONTOS PARA VISUALIZAÇÃO E INTERPRETAÇÃO DE MODELOS DIGITAIS DE AFLORAMENTOS***

**Gabriel Marx Bellina**

Apoio



São Leopoldo, outubro de 2013.

**Gabriel Marx Bellina**

**OUTCROP EXPLORER: UM SISTEMA DE RENDERIZAÇÃO BASEADA  
EM PONTOS PARA VISUALIZAÇÃO E INTERPRETAÇÃO DE MODELOS  
DIGITAIS DE AFLORAMENTOS.**

Defesa final de Mestrado apresentada ao Programa de  
Pós-Graduação em Geologia – Área de  
Concentração em Geologia Sedimentar - Linha de  
Pesquisa Sensoriamento Remoto e Modelagem Geológica,  
da Universidade do Vale do Rio dos  
Sinos (UNISINOS), como parte dos requisitos  
para a obtenção do título de Mestre em Geologia

Orientadores:

Prof. Dr. Francisco M. W. Tognoli (PPGEO-Unisinos)

Co-orientador: Prof. Dr. Mauricio R. Veronez (PPGEO-Unisinos)

São Leopoldo, outubro de 2013.

Dados Internacionais de Catalogação na Publicação (CIP)

B444o Bellina, Gabriel Marx

Outcrop Explorer : um sistema de renderização baseada em pontos para visualização e interpretação de modelos digitais de afloramento / Gabriel Marx Bellina. — 2013.

36 f. : il.

Dissertação (Mestrado em Geologia) – Universidade do Vale dos Sinos, 2013.

Orientação: Prof. Dr. Francisco M. W. Tognoli

Co-Orientador: Prof. Dr. Maurício R. Veronez

1. Geologia 2. Modelos digitais de afloramento I. Título.

CDU 55

A dissertação de Mestrado

**“OUTCROP EXPLORER: UM SISTEMA DE RENDERIZAÇÃO BASEADA EM  
PONTOS PARA VISUALIZAÇÃO E INTERPRETAÇÃO DE MODELOS DIGITAIS  
DE AFLORAMENTOS”** apresentada por **GABRIEL MARX BELLINA**

foi aceita e aprovada como atendimento parcial aos requisitos para a obtenção do grau de

**MESTRE EM GEOLOGIA** pela seguinte banca examinadora:



Prof. Dr. George Luiz Fajizotto  
Universidade Estadual Paulista Júlio de Mesquita Filho



Prof. Dr. Luiz Gonzaga da Silveira Junior  
Universidade do Vale do Rio dos Sinos



Prof. Dr. Francisco Manoel Wohnrath Tognoli  
Presidente da Banca Examinadora  
Universidade do Vale do Rio dos Sinos

São Leopoldo, 25 de outubro de 2013.

## SUMÁRIO

<b>A) TEXTO INTRODUTÓRIO .....</b>	<b>1</b>
<b>B) RESULTADOS</b>	
<b>ARTIGO SUBMETIDO PARA O PERIÓDICO <i>COMPUTERS &amp; GEOSCIENCES</i></b>	
<i>ABSTRACT</i> .....	10
1. INTRODUCTION .....	11
2. COMPUTATIONAL APPROACHES ON POINT-BASED REPRESENTATIONS .....	12
2.1. Data structure organization.....	13
2.2. View-dependent LOD-selection .....	14
3. OUTCROP EXPLORER DEVELOPMENT METHODOLOGY .....	14
3.1. Importing.....	16
3.2. Navigation.....	17
3.3. Tools .....	19
3.3.1. Edit/delete .....	16
3.3.2. Lines .....	20
3.3.3. Planes .....	21
3.3.4. Exporting.....	22
5. DISCUSSION .....	22
6. FINAL REMARKS .....	24
7. ACKNOWLEDGMENTS .....	25
8. REFERENCES .....	26
<b>C) DESENVOLVIMENTOS FUTUROS .....</b>	<b>31</b>

## Contextualização

O contexto temático que deu origem a este projeto, em 2009, era o de potencializar o uso do Laser Scanner Terrestre para aplicações em Geologia a partir do Trabalho de Conclusão de Curso de Gabriel Marx Bellina (Bellina, 2009) e da Dissertação de Mestrado de Fabiano Ferrari, iniciada também em 2009 (Ferrari, 2011; Ferrari et al., 2012). O equipamento tinha sido recém adquirido pelo Programa de Pós-Graduação em Geologia e, desde o início, ficou nítido que as pesquisas derivadas deveriam ter um viés de aplicação a partir do desenvolvimento de soluções específicas para os temas de interesse em Geologia.

Baseado na técnica LIDAR (*Light Detection and Ranging*), o equipamento é capaz de varrer e perfilar afloramentos rochosos com rapidez e obter nuvens de pontos com resolução da ordem de milímetros. Essa característica oferece aos geólogos a possibilidade de integrar dados de campo, usualmente coletados de forma analógica com equipamentos como bússola, sistemas de posicionamento GNSS, câmera fotográfica, clinômetro, martelo, trena, vara de Jacó etc, a dados digitais adquiridos por meio de sensores remotos como gamaespectrômetro, radar de penetração no solo (GPR) e Laser Scanner Terrestre (LST). Além disso, a representação digital dos afloramentos permite ampliar a qualidade do dado passível de interpretação e supre uma carência de dados na transição entre as escalas de trabalho de campo e regional (Fig. 1).

Diversas eram as frentes de pesquisa para este tema. No entanto, interagir com um modelo digital de afloramento, ter boa qualidade visual e interpretá-lo com ferramentas de cunho geológico constituía um desafio nesse momento. O grande volume de dados de uma nuvem de pontos dificultava a tarefa de visualização em aplicativos já existentes, que não dispunham de ferramentas que pudessem ser úteis à interpretação geológica. Logo, o foco do estudo foi o desenvolvimento de uma ferramenta que fosse capaz de unir ferramentas de edição e interpretação geológica, com a capacidade de gerenciar eficientemente uma grande quantidade de dados de um Modelo Digital de Afloramento (MDA).

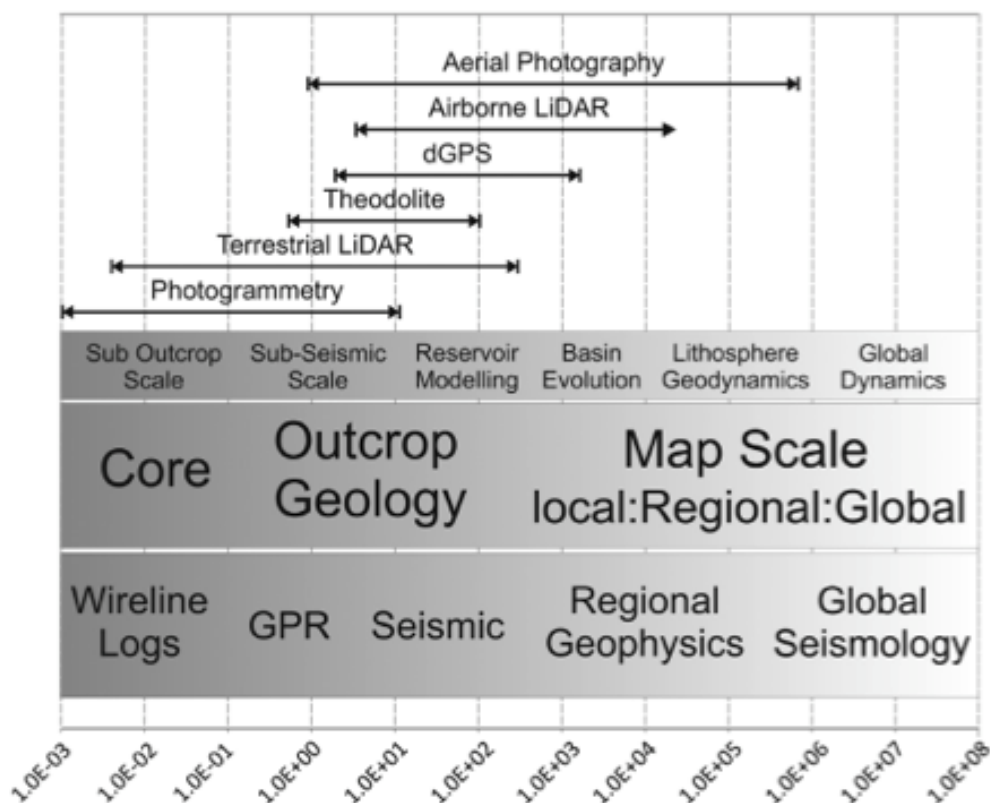


Figura 1. Comparação entre as escalas de trabalho em Geologia e aquelas atendidas por diferentes técnicas usualmente utilizadas (Hodgetts, 2013). Notar a abrangência da técnica LIDAR entre as escalas de afloramento e mapeamento regional.

## O Projeto

O projeto foi iniciado em 2009, durante o Trabalho de Conclusão de Curso em Ciência da Computação. O foco foi buscar por uma solução de visualização mais adequada do que a triangulação (*triangulation irregular network*). Devido ao grande volume de dados das nuvem de pontos resultante de imageamento de afloramentos, da ordem de dezenas a centenas de milhões de pontos, a pesquisa seguiu a linha da renderização, ou criação de imagens, usando diretamente os pontos como base, de forma a evitar a computacionalmente onerosa criação de malhas de polígonos. Os conceitos de Level of Detail (LoD) e divisão espacial hierárquica em formato de octree\* foram utilizados de forma simplificada como estruturas auxiliares à criação de níveis de resolução. Com essa abordagem foi possível visualizar uma nuvem de pontos, identificar feições geológicas (i.e., estratificações cruzadas) e navegar com relativa rapidez utilizando um *notebook* com configuração doméstica padrão, validando a utilidade da renderização por pontos para essa finalidade.

\* Octree – Estrutura hierárquica em formato de árvore na qual cada nó (não folha) é conectado a até 8 nós-filho.

A ferramenta seria mais tarde reconstruída, já no projeto de mestrado (iniciado em 2010), não mais com o foco apenas em visualização. A visualização baseada em pontos foi mantida como uma solução alternativa à tradicional triangulação mas, além disso, era necessário que a ferramenta tivesse rapidez na navegação independente do tamanho da nuvem de pontos, uma estrutura de gerenciamento de dados adequada e a integração com ferramentas de edição e de interpretação para fins geológicos. Este desenvolvimento buscou adicionar à ferramenta passos de fluxos de ações (chamados neste documento de *workflows*) previstos por artigos da área (e.g., Buckley et al., 2008), que incluem desde a aquisição dos dados pelo *Laser Scanner* até o produto final com as análises estratigráfica e estrutural do afloramento. A ferramenta deveria, então, concentrar uma série de passos desse *workflow*.

Para a correta compreensão desta tarefa, foi necessária a divisão da pesquisa em três tópicos. O primeiro era o entendimento da perspectiva do geólogo em relação à abordagem do tema e às expectativas em relação à ferramenta; o que motivou o ingresso no Programa de Pós-Graduação em Geologia (PPGEO). O segundo, a imersão no tema e a elaboração do estado da arte sobre aplicações da técnica LIDAR para fins geológicos. E o terceiro, a pesquisa das possíveis abordagens computacionais para o tema. Apenas depois disso, o desenvolvimento da ferramenta foi possível.

## **A pesquisa**

Para o entendimento da perspectiva geológica, foram escolhidas disciplinas do Programa de Pós-Graduação em Geologia focadas na análise, caracterização e interpretação de dados de campo. A percepção da natureza do dado pelo desenvolvedor, com formação em outra área de conhecimento, foi fundamental para identificar a necessidade real do usuário final da ferramenta. Em um segundo momento, foi realizada uma busca aprofundada do estado da arte do uso da técnica LIDAR para fins geológicos. Diversas são as possíveis áreas de uso deste equipamento, desde a varredura de afloramentos vulcânicos até o mapeamento de fluxos de detritos e, ainda, para propósitos educacionais. Uma linha importante seguida pelos acadêmicos foi o desenvolvimento de um *workflow* desde a aquisição até a representação do afloramento digitalizado, com as feições geológicas disponíveis para interpretação com boa qualidade.

Enquanto a técnica LIDAR se desenvolvia em outras áreas de conhecimento, como Engenharia Civil, Arquitetura, Fotogrametria, a Geologia ainda carecia de recursos que pudessem



atender suas demandas. Diversos estudos criaram soluções internas (não disponíveis para uso) e *softwares* comerciais não dispunham de ferramentas que pudessem ser utilizadas em interpretações geológicas. A interpretação de modelos digitais de afloramentos somente podia ser realizada com o uso de vários *softwares*, cada um cumprindo um requisito e necessitando de operações específicas e complexas de exportação e importação até se chegar ao modelo digital de afloramento interpretado (Bellian et al., 2005; Buckley et al., 2008; Ferrari et al., 2012). Outro ponto de discussão foi quanto às grandes nuvens de pontos geradas. Em muitos artigos científicos a solução encontrada foi descartar dados da nuvem de pontos (Bellian et al. 2005; Buckley et al. 2008; McCaffrey et al., 2009; Oryspayev et al. 2012), mas a qualidade e a resolução do MDA eram afetadas com este procedimento. Dessa forma, essa solução foi descartada para esse trabalho por não contemplar a quantidade e a qualidade do dado desejado de acordo com a capacidade de aquisição do *Laser Scanner*.

Alguns estudos descreveram, de forma não aprofundada, o desenvolvimento de suas soluções. Bellian et. al. (2005) criou um *workflow* para interpretação de MDAs, usando linguagem VRML, Polyworks®, GOCAD® e uma ferramenta desenvolvida internamente. O *workflow* tem início com a aquisição dos pontos com o *Laser Scanner* e câmeras digitais. Logo, a nuvem é integrada usando Polyworks® e então o software desenvolvido torna o modelo mais leve capacitando a interpretação no GOCAD®.

Buckley et al. (2008) também apresenta um trabalho sobre análise estratigráfica em um MDA. Neste estudo, a nuvem de pontos de adquirida com o objetivo de criar o relevo do afloramento. Depois disso, fotos do afloramento foram sobrepostas no modelo digital. O *software* Polyworks®, baseado no algoritmo ICP (Iterative Closest Point), foi usado novamente para a criação do modelo. Porém, para possibilitar a triangulação do MDA, uma grande quantidade de pontos teve que ser descartada. As fotografias foram tiradas do afloramento em pontos onde as coordenadas eram conhecidas e então, com auxílio de desenvolvimentos internos, o modelo foi completado com as imagens. O produto final possibilitou a análise visual para definição de camadas e estas foram exportadas e depois tratadas em *softwares* de modelagem de reservatório.

Outros estudos não focaram nos meios de realização da análise de MDA, mas nas aplicação e integração destas análises à área temática do estudo. Para a área educacional, McCaffrey et al. (2008) usou uma octree para filtragem e remoção de grande quantidade de dados da nuvem de pontos. Com estes dados o modelo foi gerado e sobre ele imagens foram sobrepostas. Fabuel-Perez et al. (2009)

focou na definição, medição e classificação de corpos rochosos, usando diferentes aplicativos, como RiScan Pro<sup>®</sup> and Polyworks<sup>®</sup>. Porém, apenas com o auxílio do *Virtual Reality Geological Studio*, a manipulação dos dados do modelo e a análise desejada foram possíveis. A orientação de superfícies planares foi alvo de estudo por Olariu et al. (2008). Os autores criaram ferramentas para a identificação de planos, que puderam ser identificados e orientados a partir do MDA. Ainda seguindo a linha da análise estrutural, Rotevatn et al. (2009) trabalhou o levantamento de falhas e fraturas sobre um modelo em *grid* objetivando uma precisa análise de fluxo de fluídos.

Com as necessidades do usuário avaliadas, pode-se definir os requisitos para o projeto, apresentados resumidamente abaixo:

#### 1- Importação:

A nuvem de pontos do *laser scanner* traz informações de posição (x,y,z), dados de cor RGB e intensidade. Como processo inicial, a ferramenta deve trabalhar estes dados e organizá-los em uma estrutura que possibilite um acesso otimizado.

#### 2- Criação do modelo e navegação:

A partir das nuvens de pontos importadas, o aplicativo deve criar um modelo digital que represente o afloramento. Ferramentas de navegação, como rotação, zoom e translação devem estar presentes, fornecendo um ambiente amigável ao usuário.

#### 3- Edição:

Ferramentas de edição para remoção de vegetação, obstáculos ou qualquer dado não relevante. A edição deve ser manual, com uso do *mouse* para escolha de áreas a serem descartadas.

#### 4- Interpretação:

Ferramentas de criação de linhas e planos, junto a ferramentas para medição e orientação de planos;

#### 5- Exportação:

Necessidade de exportação tanto dos dados interpretados quanto de modelos simplificados do afloramento (com menor quantidade de pontos). Adicionalmente, deve ser

possível exportar imagens ou vídeos, para finalidades diversas, especialmente para uso educacional.

Depois dessa pesquisa, o trabalho foi avaliado por uma banca no Seminário Inicial de Mestrado e a indicação de pontos a serem aprofundados sob a perspectiva computacional levou à realização de uma pesquisa mais específica sobre as possíveis abordagens para os problemas propostos. Os principais pontos foram: plataforma, algoritmos para renderização e organização dos dados da nuvem de pontos. Para a plataforma, inicialmente foi verificada a opção de incluir o desenvolvimento diretamente no GOCAD, do qual o PPGEIO é membro do consórcio. Porém, após análise da viabilidade de execução nessa plataforma e discussões internas, definiu-se que seria estratégico que toda a tecnologia envolvida fosse aberta. Desta forma, a abordagem inicial desenvolvida no Trabalho de Conclusão de Curso foi resgatada e aprimorada. Deste trabalho, foram herdadas a linguagem (Delphi) e o banco de dados (PostgreSQL). Analisou-se a possibilidade de transferir o desenvolvimento para linguagens que oferecessem maior desempenho, porém pesquisas por estudos na área se mostraram pouco conclusivos. Karac (2011) mostra um comparação de linguagens sob diversas perspectivas. A linguagem Delphi mostrou um ótimo desempenho nesta comparação, porém é fato que não está totalmente claro quais métodos foram utilizados para chegar a estas conclusões.

Por outro lado, tão importante quanto o desempenho, os grupos e comunidades de discussão somados à vasta documentação fazem da linguagem Delphi bastante acessível. Por exemplo, a documentação referente à biblioteca DglOpenGL: [wiki.delphigl.com](http://wiki.delphigl.com). Estes motivos tornam Delphi uma opção atrativa. Porém é verdade que no momento em que o foco do projeto se voltar a otimizações sobre o processamento gráfico, outras linguagens deverão entrar nos testes do estudo. É de comum uso, por exemplo, controladores de tela em linguagem C++. Como trabalho futuro, testes verificando desempenho de módulos parecem relevantes.

O banco de dados, também herdado do aplicativo já iniciado no TCC, mostrava-se como uma questão estratégica, podendo ter grande utilidade para cenários com servidores e acesso remoto. Porém, testes usando arquivos resultaram em ganho de desempenho. Os componentes usados para o acesso ao banco impossibilitavam grandes fluxos de dado binários. Isso, por outro lado, era possível com acesso direto a arquivos. Logo, a versão padrão do aplicativo deveria usar sistema de arquivo e o acesso à base ficou desabilitada (porém facilmente ativável).

Tendo a plataforma definida, o foco voltou-se à busca de algoritmos capazes de organizar grandes quantidades de dados, a fim de basear um *software* apto a proporcionar ferramentas de navegação e edição interativas. O ponto de partida foi a busca por soluções capazes de acessar uma quantidade de dados reduzidas, reduzindo ao máximo possíveis sobrecarregamentos na CPU e GPU. A organização do espaço em blocos que pudessem ser acessos separadamente foi o início da pesquisa.

Uma série de estudos traz o uso de divisão espacial e criação de estruturas hierárquicas para organização de dados de pontos. Sainz et al (2004) testou soluções de organização espacial baseadas em octrees. Unindo cada nível da hierarquia a um nível de detalhamento, o acesso à base passa a ser gradual. Level of Detail (LOD) foi foco de estudo, no intuito de encontrar o modelo que melhor atendesse a estrutura hierárquica baseada em pontos.

Os algoritmos LOD são largamente usados com foco no tratamento de malhas de polígonos. Porém, o conceito de LOD que se adequa perfeitamente ao projeto é o View-dependent LOD. Neste modelo, os dados trazidos para serem processados são os que têm maior proximidade com o observador. Trabalhando com uma estrutura hierárquica, a cada movimentação do observador, o sistema calcula a distância, verificando se é necessária uma resolução maior (ou seja, um mergulho na estrutura hierárquica buscando por um nível mais alto do LOD).

## **O desenvolvimento**

A primeira questão a ser desenvolvida era a organização dos dados na base. A estrutura de árvore era a escolha mais óbvia mas a implementação mostrou-se mais desafiadora. Uma abordagem possível era o tratamento da nuvem de pontos diretamente no banco de dados (PostgreSQL). Entretanto, o desempenho não era satisfatório, independente da otimização da manipulação. Uma melhor velocidade de acesso foi alcançada depois da criação de chaves com relação ao posicionamento. Neste modelo, valores de X, Y e Z adicionam a chave, passando por uma equação que leva em conta a localização do nodo na estrutura.

Este modelo, unido ao uso de arquivos, resultou em bom desempenho para a organização dos dados. O segundo problema enfrentado foi com o uso da OpenGL nativa. Esta biblioteca para comunicação com a placa gráfica sofre constantemente otimização e melhoramentos. O método simples de OpenGL, em que cada objeto é enviado separadamente foi um grande ponto de atraso. A

opção a isto é o uso de *buffers*, que faz com que grandes quantidades de dados sejam enviadas de uma única vez. Foi necessário mais pesquisa para adequar a ferramenta ao uso deste modelo, incluindo a escolha e substituição de uma placa que suportasse a OpenGL 1.5. O terceiro ponto estava no tratamento do LoD. O sistema deveria manter na placa gráfica apenas os pontos necessários naquele momento. Para isto, foi preciso compreender a posição do observador a cada momento. Um modo de trabalhar com isto é por meio da manipulação das matrizes da OpenGL. Com matrizes auxiliares, sempre que um movimento de rotação ou translação ocorre, os valores são guardados e com cálculos inversos, é possível definir a posição do observador. Infelizmente não foi possível encontrar bibliotecas abertas (para Delphi) que oferecessem cálculos complexos entre matrizes. Outro problema apresentado pelo Delphi foi referente à manipulação dos *buffers* e ponteiros. A quantidade de pontos a ser visualizada varia com frequência, porém o controlador de memória do Delphi apresentou problemas quanto a isso. É verdade, porém, que o ideal é usar tamanho padrão de *buffer* e adaptar seu uso.

Por fim, as ferramentas para auxílio à interpretação geológica. A base para todas foi o clique do *mouse*. Na OpenGL, uma forma exata de se definir o local clicado é criando um vetor buscando o ponto de origem na tela e mergulhando, de acordo com as matrizes já definidas, para a posição do observador. Três são as opções de clique. Ou usa-se como apoio para navegação, ou seleciona-se um cubo (da estrutura octree) para exclusão, ou seleciona-se um ponto. Tendo os pontos selecionados, é trivial definir retas ou planos. Estruturas auxiliares foram criadas para armazenar estes dados e possibilitar sua exportação. Também sobre exportação, foram criadas as opções de exportação de uma completa camada da estrutura de LOD, exportação do vídeo que captura ações tomadas no *software*, e arquivo DXF para linhas, possibilitando integração com sistemas em geral (foi realizado um teste usando Google Sketchup para a importação de linhas DXF e camada simplificada do modelo).

Com base nessa abordagem introdutória, o manuscrito do artigo científico é apresentado a seguir como parte dessa Dissertação de Mestrado, que conta ainda com um item de recomendações para trabalhos futuros.

## Referências

- Bellina, G.M. 2009. Modelagem Digital de Afloramentos a partir da renderização por pontos. Trabalho de Conclusão de Curso, Ciência da Computação, Universidade do Vale do Rio dos Sinos, 25p.
- Ferrari, F.; Veronez, M.R.; Tognoli, F.M.W.; Inocencio, L.C.; Paim, P.S.G.; Silva, R.M. 2012. Visualização e interpretação de modelos digitais de afloramentos utilizando Laser Scanner Terrestre. *Geociências*, 31(1): 79-91.
- Hodgetts, D. 2013. Laser scanning and digital outcrop geology in the petroleum industry: a review. *Marine and Petroleum Geology*, 46: 335-354.
- Karac A. 2011. Performance Comparison of Managed C# and Delphi Prism in Visual Studio and Unmanaged Delphi 2009 and C++ Builder 2009 Languages. *International Journal of Computer Applications*, 26: 9-15.
- Sainz, M.; Pajarola, R. 2004. Point-based rendering techniques. *Computers & Graphics*, 28(6): 869-879.

## **B) RESULTADOS**

ARTIGO SUBMETIDO PARA O PERIÓDICO *COMPUTERS & GEOSCIENCES*

# **OUTCROP EXPLORER: A POINT-BASED RENDERING SYSTEM FOR VISUALIZATION AND INTERPRETATION OF LIDAR DIGITAL OUTCROP MODELS**

**Gabriel Marx Bellina**<sup>a</sup>

**Francisco Manoel Wohnrath Tognoli**<sup>a,b</sup>

**Mauricio Roberto Veronez**<sup>a,b</sup>

<sup>a</sup> *PPGEO, University of Vale do Rio dos Sinos (UNISINOS), São Leopoldo-RS, Brazil, [gbellina@gmail.com](mailto:gbellina@gmail.com)*

<sup>b</sup> *Remote Sensing and Digital Cartography Laboratory (LASERCA), University of Vale do Rio dos Sinos (UNISINOS), São Leopoldo-RS, Brazil, [ftognoli@unisinis.br](mailto:ftognoli@unisinis.br), [veronez@unisinis.br](mailto:veronez@unisinis.br)*

## **ABSTRACT**

The use of LIDAR-based models for geological outcrop studies has increased in the last decade, especially in the last five years. This technique has proven its potential to represent digitally tridimensional field data collected in outcrops, increasing the quality and amount of data available for interpretation by geoscientists. Under a computational point of view, researchers have found difficulty on handling the huge amount of data acquired by LIDAR system. Under a geological point of view, researchers have found difficulty to visualize efficiently the point cloud and to convert the point cloud to high-quality digital outcrop models (DOM) with specific interpretation tools. Some research groups have developed in-house

software solutions in order to overcome the lack of commercial solutions. All solutions must consider the large database as the pain point of the project. Outcrop Explorer has been developed to manage large point clouds, to provide geological interpretation tools and to allow integration with geological applications through data exporting. In terms of software architecture, view-dependent level of detail (LOD) and a hierarchical space-partitioning structure in form of octree were integrated in order to optimize the data access and to promote a proper visualization and navigation in the DOM. Outcrop Explorer provides to geoscientists an editing tool capable to remove noise or unnecessary portions of the point cloud and interpretation tools as lines, plane identification and orientation and different exporting formats.

Key-words: Level of Detail, Octree, Terrestrial Laser Scanner, Point Clouds, OpenGL

## **1. INTRODUCTION**

The technological evolution has brought a number of advanced solutions for geological applications in the last two decades. In general, geotechnologies have permitted geoscientists acquire an increasingly amount of georeferenced digital data through use of remote sensors. The more data are collected from sensors the more resolution and size the database has and more processing is required to provide high quality images or digital models for interpreting purposes.

The LIDAR technology, especially terrestrial laser scanners (TLS), has permitted geoscientist to improve the quality of their analyses and interpretations through digital outcrop models (DOM). Although outcrops provide an intermediary approach between work scales, i.e., 3D indirect seismic data and 1D direct well data, outcrop data acquisition is often underexplored. Considering that seismic data have their resolution limited to meters and wells are laterally restricted to few centimetres, digital outcrop models have become an interesting way to integrate surface and subsurface data. This scale of work allows



acquiring data using remote sensors, as laser scanners, as well as acquiring data using the standard field equipment.

The high accuracy of the DOMs allows recognizing and interpreting geological features within a 3D-realistic scenario (Pringle et al., 2004b; Bellian et al., 2005; 2005; Hodgetts et al., 2007; Sagy et al., 2007; Buckley et al., 2008a; Rotevatn et al., 2009). However, the huge point clouds and the high quality of the model required to represent geological data have been a challenge in this area of knowledge. The lack of integration among Geology, Geomatic and Computer Graphics has led researchers either to use non-specific applications to perform geological interpretations or to adapt interpretative routines considering the available software solutions, many times discarding data from the point cloud.

In-house computational systems have been developed in the last years by some research groups specifically to handle the LIDAR data and to provide efficient visualization and some specific geological interpretation possibilities (Bellian et al., 2005; Olariu et al., 2008; Hodgetts et al., 2007; Fabuel-Perez et al., 2009). But there still is a lack of application that could manage the end-to-end DOMs evaluation processes. The integration among the well-known visualization technique level of detail (LOD) together with an efficient data management and some editing and interpretation tools has permitted the development of the Outcrop Explorer, a specific application for point cloud and DOM visualization, manipulation and interpretation capable of exporting in different formats for subsequent modelling activities.

## **2. COMPUTATIONAL APPROACHES ON POINT-BASED REPRESENTATIONS**

The concept of scene building using points as base is a recurrent topic on Computer Graphics. Levoy & Whitted (1985) initially applied this concept for rendering surfaces and since the 2000's more studies have been performed (Zwicker et al., 2001; Sainz & Pajarola, 2004). The development of laser scanners capable to acquire huge point clouds and their subsequent utilization for acquiring data from geological

outcrops has created the necessity of solutions to handle the database obtained by this technique as well as to have good-quality in the visualization of the DOMs.

## 2.1 Data structure organization

A common approach is through hierarchical space-partitioning data (Levoy et al., 2000; Ren et al., 2002; Botsch et al., 2002; Botsch et al., 2003). This model is based on building different LODs for a same model in which a tree structure will have each depth level assigned to one of these LODs (Fig 1). Good performance was achieved after building this architecture based on octrees (Sainz et al., 2004).

The octree concept was developed by Meagher (1980), published and patented respectively in 1982 and 1987 (Meagher, 1982,1987) and many different applications for 3D representation have been developed since then (Ren et al., 2002; Pfister & Zwicker, 2002; Botsch, 2002,2003; Pajarola, 2004; Eberhardt et al., 2010). It is based on a hierarchical tree data structure in which each node can have either eight children or no children. Each node can once more be divided in eight parts, and so forth. The subdivision based on a factor of 2 allows represent the three dimensional space with an increased voxel resolution. Partitioning, downsampling and search operations on the point data set are some of the resources allowed from this approach.

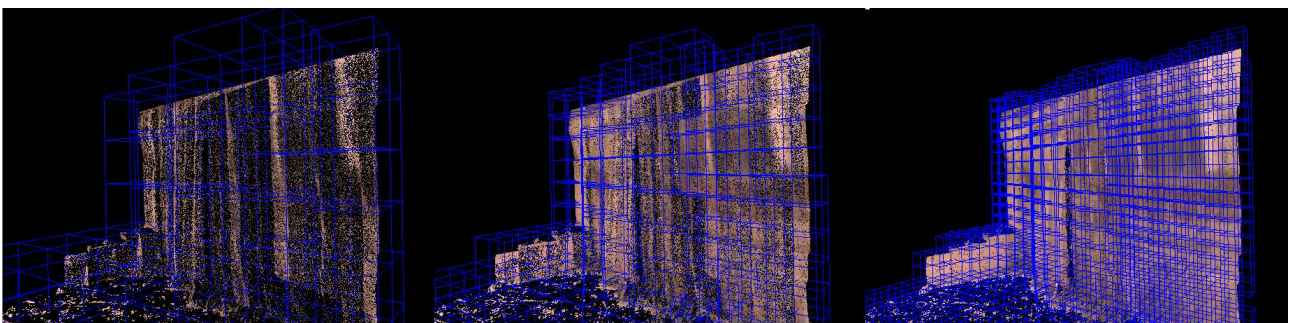


Figure 1. Example of hierarchical spatial subdivision of an adaptive point-octree.

## **2.2 View-dependent LOD-selection**

The standard algorithm for LOD is based on building different degrees of resolution and, at a specific moment, the access only to the necessary data is done. The amount of data may depend of its application, like the viewer speed or distance. Luebke et al. (2003) present a number of approaches and challenges for this technique, including a case study that used meshes as base. For the context of view-dependent LOD based on hierarchical spatial-partitioning of point clouds, the hierarchy is subject to a check of distance. Every time a node is defined as too close from the viewer, its children will be brought to the same check, forcing more resolution to be displayed.

Another example of view-dependent LOD was implemented by Rusinkiewicz & Levoy (2000). They developed a tool called Qsplat to render point clouds with a dynamic adaptation of the LOD, in which used a hierarchical LOD for storing the point cloud on a pre-processing step. The process is based on spheres. The algorithm first identifies cluster of points and create spheres joining the points on an object. Later, bigger spheres are built joining smaller spheres. After this pre-processing, the Qsplat is ready to render the object. The program takes the highest spheres on the structure and check if they are on the screen. If not, all the smaller spheres that are inside of this one are also refused. But if yes, the object is displayed and all the smaller spheres inside are put under the same process done with the bigger spheres. This algorithm also provides a mechanism to prevent slowness by not going deeper on the structure if this condition is detected. In order to improve the technique, Wand et al. (2008) developed an application capable to handle and edit huge point clouds. The concept of octree was adapted in order to accept dynamic editing, going deeper while optimizing the inevitable out-of-core storage.

## **3. OUTCROP EXPLORER DEVELOPMENT METHODOLOGY**

The starting point for developing this project was to find a solution for handling large amount of data. A view-dependent LOD together with a spatially partitioned hierarchical structure was the best choice

in this case. The LOD concept plus the adopted octree structure allow sending to the graphic card only the relevant data and to split the access to the hard disk, which is a pain point in this process. This approach served as base for the navigation and some of the tools built subsequently.

To properly define and document the steps required to analyse and process a DOM, from the point clouds to the interpreted data, some workflows have been proposed (Bellian et al., 2005; Enge et al 2007; Buckley et al., 2008; Alfarhan et al., 2008; Fabuel-Perez et al., 2010; Ferrari et al., 2012) as well as the workflow adopted herein (Fig. 2). Common steps involve data acquisition, DOM creation, noise/obstacle editing, geological interpretation and exporting of interpreted data. A common characteristic among the different workflows is the necessity of using different software products, each of them specific for different tasks, that increases the time of processing and interpretation, decreases the visual quality of the DOMs in manipulation steps and might face difficulties on data integration among systems.

The necessity of a tool capable to integrate several steps among the data acquisition and the interpretation of a DOM motivated this project. Outcrop Explorer is a software solution developed to process, edit, interpret and export data and interpretations derived from LIDAR systems (Fig 2). The following requirements have been addressed: a) rendering of DOM from LIDAR point clouds; b) manual noise/obstacle editing; c) manual geological interpretation tools, like lines and plane orientation and d) data export. After having this structure as base, three modules were defined to split the entire process of handling the database: importing, navigation and editing tools.

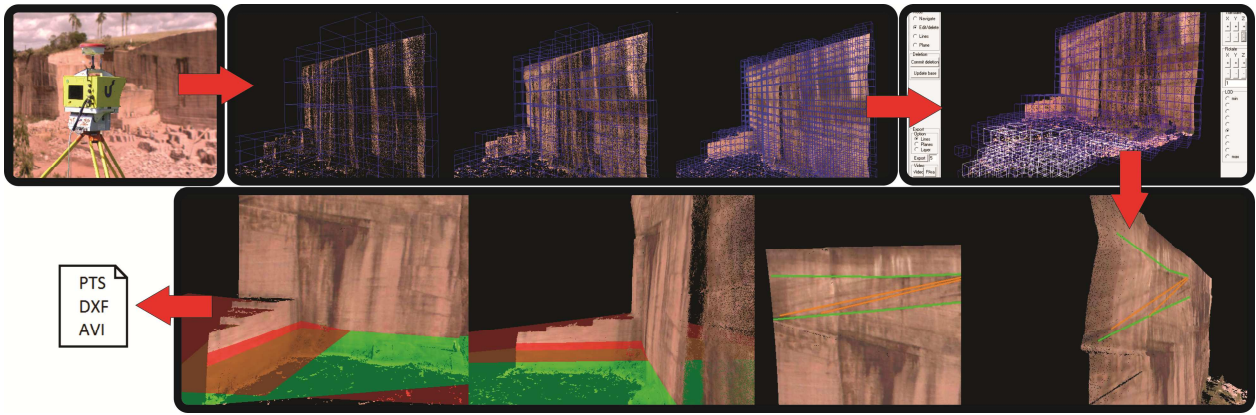


Figure 2. From the outcrop to the digital outcrop model workflow adopted in this study. a) The outcrop. b) The point cloud acquired by laser scanner. c) Automatic hierarchical structure enhanced by the blue cubes. d) Obstacle and noise editing. White cubes selected for removing unnecessary data. e) Interpretation tools; lines demarcating stratifications (see the same lines in different viewing) and stratification planes defined (arrows indicate the three points used for defining the plane). f) Data exporting formats.

### 3.1 Importing

This module reads the point cloud from a file creates the hierarchical structure and stores it on the disk. For storing the structure, the most used solution is by simple file system. To build the solution using files offers great possibilities, since it allows the developers to speed up the data reading by several ways. Pajarola et al. (2005) used files to use a pre-defined stream of data.

The hierarchical space partitioned structure used for this project is based on a simple octree. Each tree depth level is correspondent to a level of resolution built. It comprises two processes: one for building the structure frame and another for building the LODs. The first process defines the number of layers or the hierarchical structure depth. Two values are used, the first is the smallest spatial cube, based on a constant (which can be modified by the user) and the second is calculated based on the distance between the farthest points, which have the XYZ coordinates as comparison parameters. With these data, a giant cube is calculated. This cube should contain all points and its side size will be defined by the following equation:  $BCS = (SCS) \cdot 8^L$ , where BCS = biggest cube size, SCS = smallest cube size, L = number of layers.

As soon as the biggest cube is defined, the space inside this cube is divided in eight. Each of these divisions will be called child of the biggest cube. Again, each cube is subdivided in eight and the process will repeat until the number of layers is reached. Each cube represents a node on the hierarchical structure. On a subsequent check, all the cubes that don't contain points inside its area are deleted from the structure, together with its subdivision.

As a second process, the LOD is built by a down-top perspective. First, each point from the point cloud has its position evaluated. The position will define in which node of the octree the point should be stored. This process is optimized using a position-based key, built with power of 2 series (where x, y and z use different powers). For storage, the application was built to work with DataBase (PostgreSQL) or system files.

### **3.2 Navigation**

After the hard disk octree structure is built and filled the displaying module takes place (Fig. 3). System builds an in memory octree, based on the hard disk already created structure. But the point cloud is not yet taken. The behavior of the system is basically divided on two steps. One of them will manage what data will be sent to the graphic card, whereas the other will manage the data that still need to be taken from the database. There is an initial data load from the database, that can take small amounts of data or it can take all database, depending of the size of the database. It is given to the user the option of choosing which LOD will be used in the first data load, since specific tasks may not require much detail. Even though, it is recommended to use a level close to the maximum number of layers.

After that, system starts managing how much data will be sent to render. As above-mentioned, this evaluation is done by calculating each node on the octree with the viewer position. If a specific node is too far, no much detail is required, and therefore there is no necessary to dig into the structure for that area/node. For closer nodes, more detail is required and the data is acquired going deeper and deeper on

the structure for that node. The nodes that are outside of the screen are not considered. Figure 4 shows the system workflow shows the DOM after the initial processing using the cube structure with different cube sizes calculated from the hierarchical octree structure.



Figure 3. Outcrop Explorer's visualization and navigation module. Note the high quality of the sandstone wall as evidenced by the crossing lines (cross-stratifications) in the upper left corner of the point cloud.

The detail to be achieved depends on the user requirements at the moment. There is an option to increase or decrease the factor used for calculating the distance. This might cause slowness when the maximum level is required; therefore, a slight reduction of the resolution is of great help while navigating. After defined the area to be interpreted, navigation will not be necessary and resolution can be increased.

For handling the overall communication with the graphics card interface, the API chosen was OpenGL, which offers all necessary tools to render the created outcrop model and has an extensive documentation available. With help of OpenGL, a process updates the GPU memory (Graphics Processing Unit) with the list of points currently on the main memory. The transmission of data from CPU to GPU memory is a possible bottleneck. Therefore, it was chosen to use VBO (Vertex Buffer Object) to converge the efforts (Scheiblauer, 2006). The usage of buffer objects enables a great amount of data to be transferred to the graphic card memory on a single call. The call is made always after the viewer position or

the LOD parameter is changed. After a process showed on Figure 4, a VBO is created to manage the data transmission.

Now, as a second step, if the nodes required are not on the main memory, which means that they were not taken at the first load mentioned above, the system will have to look for this information on the database. Since this access is time consuming, a buffer environment was created where each node accessed will have its points stored on the main memory, based on a parameter entered by the user.

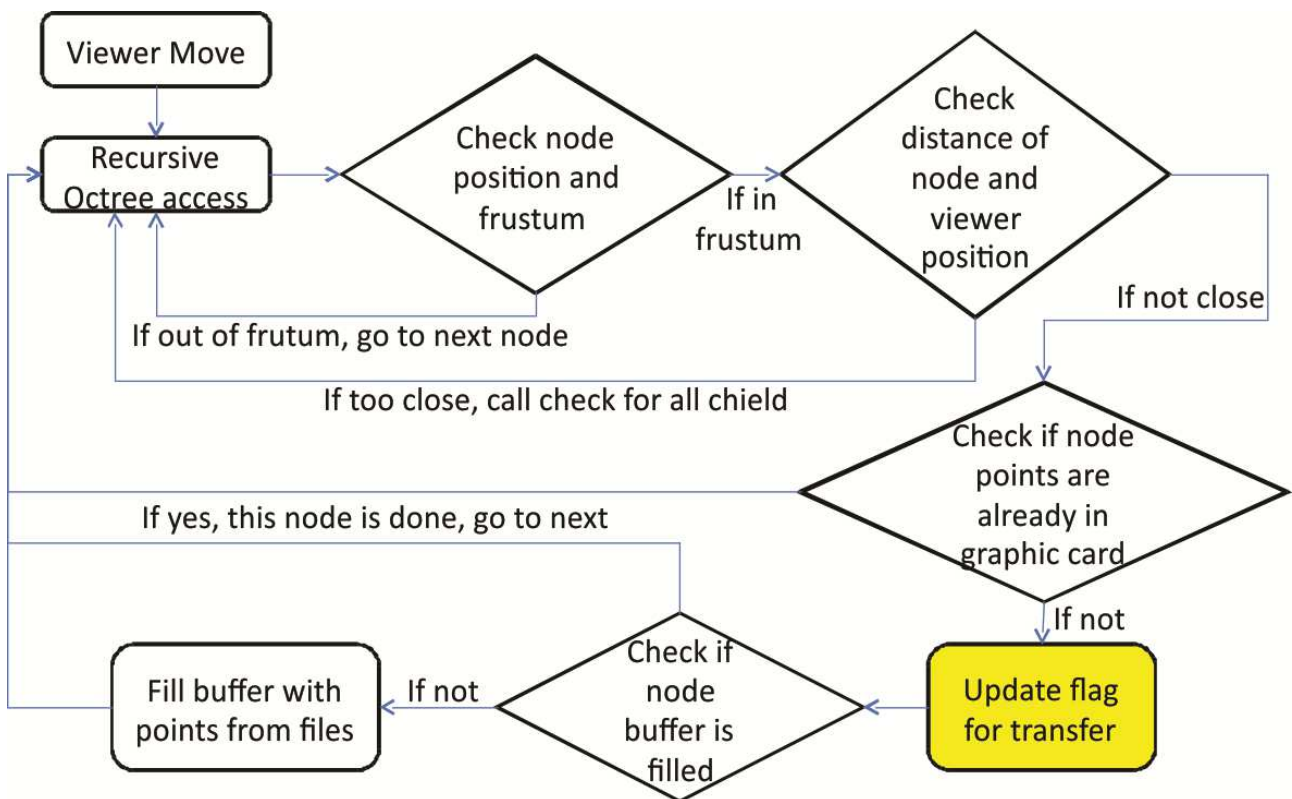


Figure 4. Logic workflow for view-dependent LOD used in Outcrop Explorer.

### 3.3 Tools

With the navigation properly addressed, the next step is to offer tools that would enable geoscientists interpret outcrops as presented in DOMs. Most options require the ability of finding the place on OpenGL respective to the mouse click place. The most useful tools for this application are: remove



manually irrelevant data, create lines, create planes using three points as base and export the interpretations in different formats, like .dxf, .avi and .pts.

### 3.3.1 Edit/delete

During acquisition by laser scanner, it is very common the presence of some type of obstacle between the equipment and the target, such as plants and random objects. This type of data is irrelevant under a geological perspective but increase a lot the amount of data to be managed in the database. To remove this data, it was created an option using the octree structure. The user has the option to click at any point and the system will identify the current octree node that those points belong (Fig. 5). Then, the user can remove this node and its children from the current visualization or it can be deleted directly from the database. This second option will also require a reconstruction of the structure, to remove the points also from the parent nodes. However, this process can be done later and navigation and interpretation are not prejudiced.

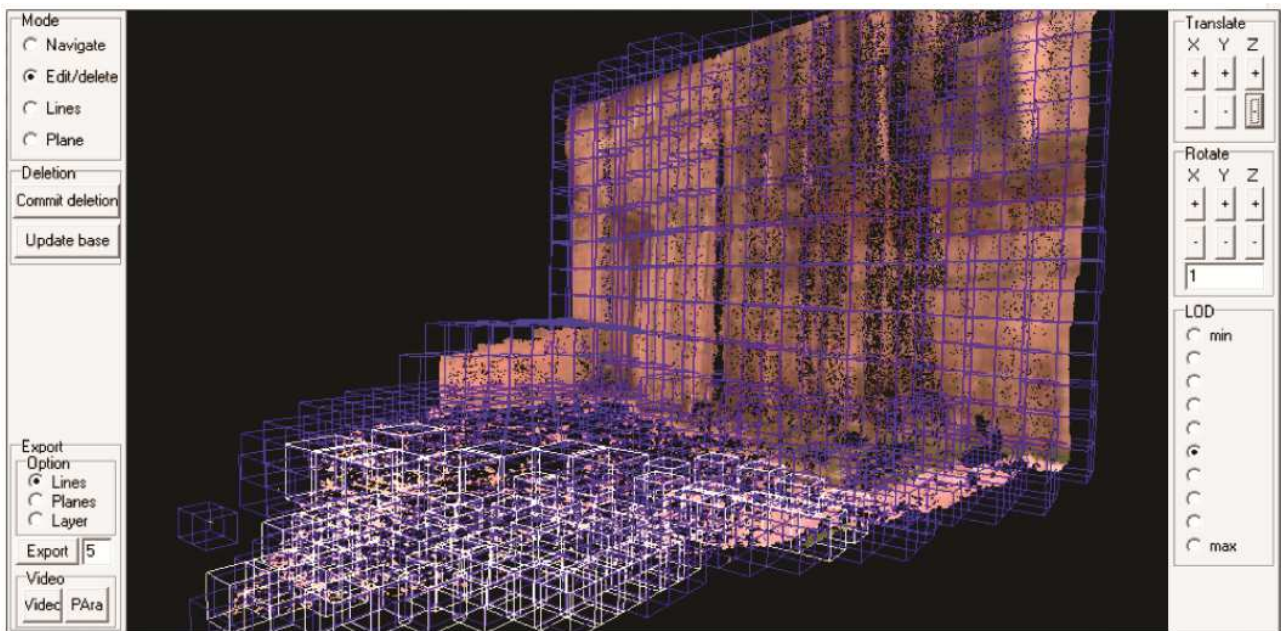


Figure 5. Digital outcrop model (DOM) with the edition mode activated. Blue cubes are those that user may choose to delete. The white ones were manually selected to be deleted of the octree. Note that relevant data are in the vertical wall (sandstone); the selected white cubes represent the floor of the quarry and other types of obstacles.

### 3.3.2 Lines

With the lines option, the interpreter can identify stratifications, beddings, fractures, faults, geometries and any kind of geological features relevant for interpretation from mouse clicks on the screen (Fig. 6). The tool also calculates the line size according to the coordinates.

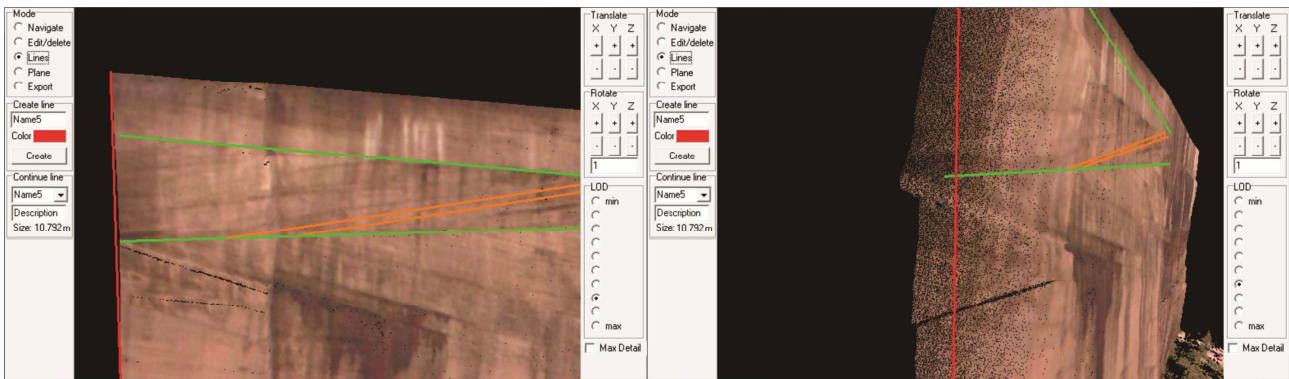


Figure 6. Interpretation lines revealing a cross-stratification directly on the DOM. Colors can be changed to enhance hierarchy or specific features. Note that lines are attached with the point as shown by the rotation of the DOM.

### 3.3.3 Planes

Planes are almost always difficult to measure in outcrops due to the lack of good exposures in which geoscientists can take a great number of measures using a compass. In vertical outcrops, access in the upper levels is quite dangerous or even impossible. However, data from point clouds can easily provide a set of planes of the entire outcrop. Different methods can estimate the orientation of planes from point clouds, such as three points, planar regression and inertia moment (Souza et al., 2013). Outcrop Explorer is capable to create a plane using any three points on the screen (Fig. 7). The software will define and show it on the screen with an alpha factor (transparent effect). Therefore, it was also created an option to set the north direction in the tool and, consequently, the system automatically calculates the spatial orientation of any plane.

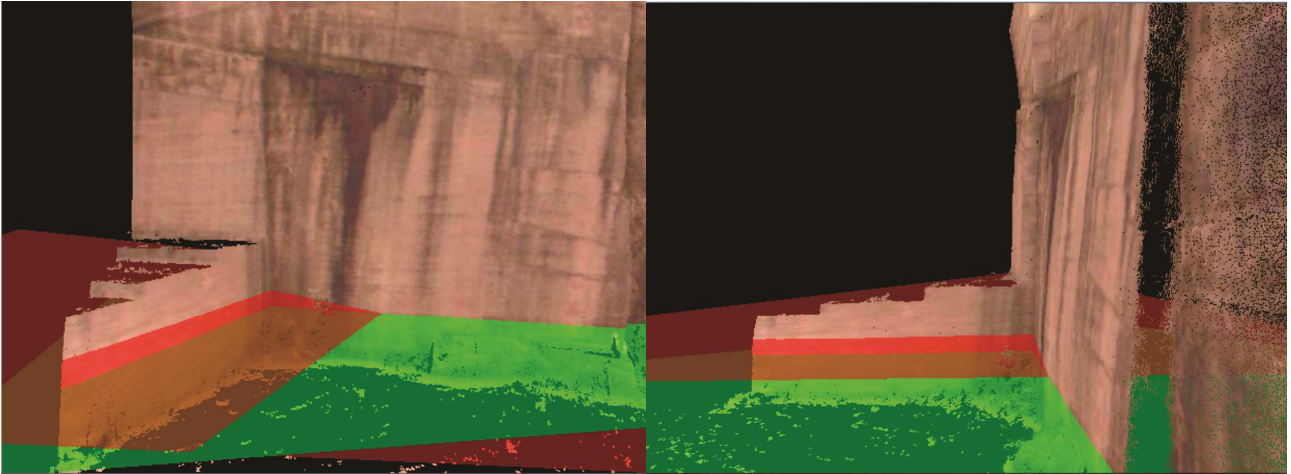


Figure 7. Plane created from three clicks directly on the digital outcrop model as seen on the screen.

### 3.3.4 Exporting

As soon as the interpretation is created, it is possible to export it. The system can export the lines, the planes or any specific layer of the digital outcrop model as defined by the LOD. Most of the software products used for these purposes are not capable to handle large database, especially the CAD-based applications. Thus, it seems to be a good option to export a layer from the DOM with fewer amount of data after the interpretation has been performed with the adequate quantity of data.

After interpreted, the model can have the data exported. The system allows creating PTS files, which can be imported in CAD or GIS-based applications. Another possibility for educational purposes or even for electronic publications is to export a video from the system. Currently, only short videos with less than a minute can be exported, once that system uses AviWriter and as this component works with BMP images, the output video easily reaches a large file size.

## 4. DISCUSSION

During the development, performance issues needed to be constantly addressed. For data access, both database and system files were tested. System files solution presented a faster data retrieval rate and

was defined as the standard option, but for further developments, database can be an interesting option in case of mobile access, therefore it remains available. Optimization on file access, like data compressing or stream of data storage can provide performance boost. For this project, the files are stored from a binary memory stream representing a buffer of points. This reduces the time processing for each file; they are simply transferred to a memory variable.

There are patterns of access that can be observed, like usually when a node is accessed, the closest ones will also be required. These patterns can be used to optimize the data access. Huang et al (2006) presented an application that goes on this direction and it can be interesting for future implementation on the developed tool.

To handle the navigation, while good amount of data is already in the main memory, an octree structure is used for frustum culling. Together with the View dependent LOD, only the necessary data is send to the Graphic card at any movement. But even this amount can be too much at certain conditions or may not be necessary for the user at some specific moment. Therefore, it was created a degree option. User can easily reduce the level of detail an any moment, to avoid performance issues, and raise it again at any moment that interpretation is to be done.

To test the solution, two point clouds were used. One acquired on the Taquara's sandstone quarry and a second on the Morro do Papaleo outcrop. The DOMs created by the tool was then subject of the interpretation features. The figures 5, 6 and 7 were the result of a manual use of it. After creating it, an integration test was done with a generic spatial editing software: Google Sketchup. A pts file was created with a simplified version of the point cloud and for the lines, a dxf file. The software was able to import and visualize the exported data.

The Taquara's outcrop point cloud has 4,786,072 points. Working with high level of detail, the navigation was always handled with no less than 4 frames per second (fps) for a close view. The editing

option used to wipe off the ground and some vegetation present in the point cloud helped to increase this number considerably and achieve a proper navigation.

The second point cloud was that from Morro do Papaleo Outcrop. The number of points of this outcrop is 56,612,473. Approximately 6 million points were eliminated from the point cloud because they represented obstacles and vegetation in front of the outcrop. The navigation with low-to-moderate resolution, i.e., until layer 8 (of 11) did not have delay; however navigation reached some delay in moments of buffering or when resolution was very high. To prevent this situation, user can decide to:

(a) Have a rapid navigation with relatively low resolution or have a temporarily slow navigation for inspecting details of the DOM with high resolution (in both cases all data of the point cloud are available in the database). For example, general measurements such as height and lateral extension, delimitation of rock body geometry and interpretation of line-based structures and features can be done with the first option (see Fig. 6).

(b) Inspect a specific region of the point cloud or DOM with high resolution by removing temporarily lateral portions using the cube structure and working only with that specific region of the DOM. This possibility can be used for detailed analyses, such as those involving inspection of high detailed features or structures and selection of specific points to define and orientate planes (see Fig. 7).

## **5. FINAL REMARKS**

The purpose of this project was to develop a specific system for geoscientists to visualize point clouds of outcrops acquired using laser scanners and interpret resultant DOMs with geologically-oriented tools. Considering that many solutions for this type of application are present only in in-house developments and that geological interpretation tools are not available in the most of commercial software

products, Outcrop Explorer intends to be the beginning of a long-term project in which users around the world will be the main contributors for its development.

A lot of challenges are present in solutions like Outcrop Explorer. Under a computational point of view, there is no novelty in terms of visualization and creation of algorithms. Instead, this project is the application and combination of algorithms to create a single software solution capable to solve basic interpretation problems of the geoscientists, dedicated to represent geological data of outcrops using remote sensors as laser scanners.

Performance in visualization will always be the big deal once a huge database is a prerequisite for digital outcrop representations based on LIDAR technique. New data management structures, high performance computing and new techniques for 3D visualization are required for efficiently improve the quality of a DOM, considering that new technologies on graphic computing arise daily. We have started a long-term project that in this first approach aimed to provide a solution for visualizing and performing basic geological interpretations in Digital Outcrop Models.

## **6. ACKNOWLEDGMENTS**

The authors wish to thank the Remote Sensing and Digital Cartography Laboratory (LASERCA/UNISINOS) for facilities and equipment, especially associate technicians and researchers Marcelo Kehl de Souza, Leonardo Inocencio, Marcos Turani, Evandro Kirsten, Beto Reis and Rudi Cesar Comiotto Modena for their support in field and lab activities. Ubiratan Faccini, Kleinner Farias and Luiz Gonzaga da Silveira Jr. contributed with ideas and suggestions that greatly improved a previous version of this manuscript. GMB thanks the Bureau for the Qualification of Higher Education Students (CAPES) for providing the master's PROSUP scholarship. The project was financially supported by FAPERGS (Edital 01/2010 – Processo 10/0477-0) granted to FMWT; PETROBRAS - Technological Network on Sedimentology

and Stratigraphy (Convênio 16 – SAP 4600242459) and FINEP (Project Moda – MCT/FINEP – Pré-Sal Cooperativos ICT – Empresas 03/2010) granted to MRV.

## 7. REFERENCES

Alfarhan, M., White, L., Tuck, D., Aiken, C. 2008. Laser rangefinders and ArcGIS combined with three-dimensional photorealistic modeling for mapping outcrops in the Slick Hills, Oklahoma, *Geosphere*, 4, 576-587.

Bellian, J.A., Kerans, C., Jennette, D.C. 2005. Digital Outcrop Models: Applications of terrestrial scanning LIDAR technology in stratigraphic modelling. *Journal of Sedimentary Research*, 75, 166–176.

Botsch, M., Kobbelt, L. 2003. High-quality point-based rendering on modern GPUs. In: *Proceedings Pacific Graphics 2003*, IEEE Computer Society Press, p. 335–343.

Botsch, M., Wiratanaya, A., Kobbelt, L. 2002. Efficient high quality rendering of point sampled geometry. In: *Proceedings Eurographics Workshop on Rendering*, p.53–64.

Buckley, S.J., Howell, J.A., Enge, H.D., Kurz, T.H. 2008. Terrestrial laser scanning in geology: data acquisition, processing and accuracy considerations. *Journal of the Geological Society of London*, 165, 625-638.

Eberhardt, H., Klumpp, V., Hanebeck, U.D. 2010. Density Trees for Efficient Nonlinear State Estimation, *Proceedings of the 13th International Conference on Information Fusion*, Edinburgh, United Kingdom, July, 2010. Available at: [http://isas.uka.de/Publikationen/Fusion10\\_EberhardtKlumpp.pdf](http://isas.uka.de/Publikationen/Fusion10_EberhardtKlumpp.pdf)

Enge, H.D., Buckley, S.J., Rotevatn, A., Howell, J.A. 2007. From outcrop to reservoir simulation model: workflow and procedures. *Geosphere*, 3, 469-490.

Fabuel-Perez, I.; Hodgetts, D.; Redfern, J. 2009. A new approach for outcrop characterization and geostatistical analysis of a low-sinuosity fluvial-dominated succession using digital outcrop models: Upper Triassic Oukaimeden Sandstone Formation, central High Atlas, Morocco. *AAPG Bulletin*, 93, 795-827.

Fabuel-Perez, I.; Hodgetts, D.; Redfern, J. 2010. Integration of digital outcrop models (DOMs) and high resolution sedimentology - workflow and implications for geological modelling: Oukaimeden Sandstone Formation, High Atlas (Morocco). *Petroleum Geoscience*, 16, 133-154.

Ferrari, F., Veronez, M.R., Tognoli, F.M.W., Inocencio, L.C., Paim, P.S.G., Silva, R.M. 2012. Visualização e interpretação de modelos digitais de afloramentos utilizando Laser Scanner Terrestre. *Geociências*, 31, 79-91.

García-Sellés, D., Falivene, O., Arbués, P., Gratacos, O., Tavani, S., Muñoz, J.A. 2011. Supervised identification and reconstruction of near-planar geological surfaces from terrestrial laser scanning. *Computers & Geosciences*, 37, 1584-1594.

Hodgetts, D., Gawthorpe, R.L., Wilson, P., Rarity, F. 2007. Integrating digital and traditional field techniques using virtual reality geological studio (VRGS). Society of Petroleum Engineers – 69<sup>th</sup> European Association of Geoscientists and Engineers Conference and Exhibition, 83-87.

Hunt, D.W., Gilsepie, P., Thurmond, J., Casini, G., Monsen, E. 2010. Development and testing of new automated methods for the capture of quantitative fracture data from outcrop analogues. AAPG Annual Convention and Exhibition, New Orleans, Louisiana.

Levoy, M., Whitted, T. 1985. The use of points as display primitives. Technical Report - TR 85-022, The University of North Carolina at Chapel Hill, Department of Computer Science, 13p. Available at <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.89.1180&rep=rep1&type=pdf>

Levoy, M., Pulli, K., Curless, B., Rusinkiewicz, S., Koller, D., Pereira, L., Ginzton, M., Anderson, S., Davis, J., Ginsberg, J., Shade, J., Fulk, D. 2000. The Digital Michelangelo Project: 3-D Scanning of Large Statues. In:



Proceeding SIGGRAPH'00, 14p. Available at <http://graphics.stanford.edu/papers/dmich-sig00/dmich-sig00-nogamma-comp-low.pdf>

Luebke, D., Reddy, M., Cohen, J., Varshney, A., Watson, B., Huebner, R. 2003. Level of Detail for 3D Graphic. Morgan Kaufmann as part of their series in Computer Graphics and Geometric Modeling, Morgan Kaufmann, 390p.

McCaffrey, K.J.W., Feely, M., Hennessy, R., Thompson, J. 2008. Visualization of folding in marble outcrops, Connemara, western Ireland: an application of virtual outcrop technology. *Geosphere*, 4, 588-599.

Meagher, D. 1980. Octree Encoding: A New Technique for the Representation, Manipulation and Display of Arbitrary 3-D Objects by Computer. *Rensselaer Polytechnic Institute* (Technical Report IPL-TR-80-111).

Meagher, D. 1982. Geometric modeling using octree encoding. *Computer Graphics and Image Processing*, 19,129-147.

Meagher, D. 1987. High-speed image generation of complex solid objects using octree encoding. United States Patent Office, EP 0152741 B1. Available at:

<https://docs.google.com/viewer?url=patentimages.storage.googleapis.com/pdfs/US4694404.pdf>

Olariu, M. I., Ferguson J.F., Aiken C.L.V., 2008, Outcrop Fracture Characterization Using Terrestrial Laser Scanners: Deepwater Jackfork Sandstone at Big Rock Quarry, Arkansas, *Geosphere*, 4, 247-259.

Oryspayev, D., Sugumaran, R., De Groote, J., Gray, P. 2012. LIDAR data reduction using vertex decimation and processing with GPU and multicore CPU technology. *Computers and Geosciences*, 43, 118-125.

Pajarola, R. 2005. Stream-processing points. In: Proceedings IEEE Visualization, 10p. Available at <http://www.ifi.uzh.ch/vmml/publications/older-pulications/PStream.pdf>

Pringle, J.K., Westerman, A.R., Clark, J.D., Drinkwater, N.J., Gardiner, A.R. 2004. 3D high-resolution digital models of outcrop analog study sites to constrain reservoir model uncertainty: an example from Alport Castles, Derbyshire, UK. *Petroleum Geoscience*, 10, 343-352.

Ren, L., Pfister, H., Zwicker, M. 2002. Object space EWA surface splatting: a hardware accelerated approach to high quality point rendering. *Computer Graphics Forum*, 21, 461-470.

Rotevatn, A., Buckley, S.J., Howell, J.A., Fossen, H. 2009. Overlapping faults and their effect on fluid flow in different reservoir types: a LIDAR-based outcrop modeling and flow simulation study. *AAPG Bulletin*, 93, 407-427.

Rusinkiewicz, S.; Levoy, M. (2000). QSplat: A multiresolution point rendering system for large meshes. SIGGRAPH 2000, New Orleans, Louisiana. Available at: <https://www.cs.princeton.edu/courses/archive/spring01/cs598b/papers/rusinkiewicz00.pdf>

Sagy, A., Brodsky, E. E., Axen, G.J. 2007. Evolution of fault-surface roughness with slip. *Geology*, 35, 283–286.

Sainz, M., Pajarola, R. 2004. Point-based rendering techniques. *Computers & Graphics*, 28, 869-879.

Sainz, M., Pajarola, R., Lario, R. 2004. Points reloaded: point-based rendering revisited. In: Alexa, M., Rusinkiewicz, S. (ed.). *Proceedings of the Eurographics Symposium on Point-Based Graphics*, 121–128.

Souza, M.K., Veronez, M.R., Tognoli, F.M.W., Silveira Jr., L.G., Inocencio, L.C., Silva, R.M., Modena, R.C.C. 2013. Terrestrial Laser Scanning: Application for measuring of structures information in geological outcrops. *International Journal of Advanced Remote Sensing and GIS*, 2, 260-270.

Zwicker, M., Pfister, H., Van Baar, J., Gross, M. 2001. Surface splatting. In: SIGGRAPH'01 - Proceedings of the 28th annual conference on computer graphics and interactive techniques, ACM Press, New York, 371–378.

Wand, M., Berner, A., Bokeloh, M., Jenke, P., Fleck, A., Hoffmann, M., Maier, B., Staneker, D., Schilling, A., Seidel, H. 2008. Processing and interactive editing of huge point clouds from 3D scanners. *Computers & Graphics*, 32, 204-220.

## **C) DESENVOLVIMENTO FUTURO**

A implementação do Outcrop Explorer teve um escopo restringido pelo cronograma do curso de mestrado. O planejamento da ferramenta foi realizado a partir de uma estratégia que possibilitasse a manipulação de grandes nuvens de pontos e modelos digitais de afloramento, a visualização desta nuvem e/ou modelos em uma plataforma 3D e ferramentas de edição e interpretação geológica. Durante o desenvolvimento, itens adicionais foram adicionados, como otimizações de acesso à base e transferência de dados para a placa gráfica, ou a exportação de vídeos. Porém, desde o início, o desenvolvimento objetivou servir de base para um projeto de longo prazo, onde novos requisitos pudessem ser adicionados. Três são as principais linhas de desenvolvimentos futuros: otimização, funcionalidades e plataforma.

### **Otimização**

A pesquisa computacional constantemente conclui novos estudos sobre otimização para processamento gráfico. Duas principais frentes parecem mais interessantes. A primeira é a otimização do acesso à base. Huang et al. (2006) propõe a criação de codificadores que otimizam o armazenamento e acesso a dados de nuvem de pontos. Pajarola et al. (2005) prepara o acesso a nuvem de pontos em uma série organizada. O Outcrop Explorer tem os dados organizados em um formato binário interno ao sistema, o que oferece a otimização na linha da proposta de Huang et al. (2006); porém uma pesquisa profunda sobre esta abordagem junto com as propostas de Pajarola et al. (2005) pode servir de base para um próximo projeto de pesquisa.

A segunda frente refere-se às otimizações no acesso e no uso da placa gráfica e GPU. Kovac *et al.* (2010) apresenta um projeto, assim como no Outcrop Explorer, em que existe um sistema de organização dos dados, porém existe um trabalho extra em cima dos pontos, criando área em torno dos pontos e eliminando espaços em branco.

### **Funcionalidades**

As ferramentas iniciais desenvolvidas foram baseadas nas necessidades básicas de interpretação geológica. Por exemplo, a identificação de camadas ou fratura e falhas e, logo, sua

marcação e delimitação (por interação com usuário). Depois de marcados e definidos, a interpretação pode ser exportada em formato pts, dxf ou avi. O primeiro ponto para um desenvolvimento futuro está na identificação automatizada de padrões, incluindo a orientação automatizada de estruturas planares por diferentes métodos e que tomar por base o artigo recentemente publicado por Souza et al. (2013). Certamente, requererá bastante análise pois estratificações poderão ser confundidas com marcas de intemperismo, feições lineares e planares (ex: fraturas e falhas) a partir da resposta de um algoritmo de identificação de padrões de imagem. Possivelmente, o reconhecimento automático deverá ser seguido de um processo supervisionado, em que a confirmação do usuário seja fundamental. Outra área de análise está no trabalho dos dados de intensidade do Laser. Cada ponto adquirido possui um valor de intensidade do pulso que retorna do alvo e que já se mostrou de interesse para a avaliação de características físicas e químicas do alvo (Inocencio, 2011; Bordin et al., *in press*). O trabalho de Burton et al. (2011) pode servir de base para abordagens utilizando o registro de intensidade de retorno do pulso *laser*.

Além disso, é essencial que a ferramenta seja utilizado por geocientistas para que uma avaliação crítica seja feita e que as sugestões sejam avaliadas para implantação ou correção.

## **Plataforma**

A ferramenta funciona executando em um sistema e lendo arquivos acessíveis ao usuário do sistema operacional. Porém, tem configurado o acesso ao banco de dados PostGreSQL, que possibilita o acesso remoto. Parece interessante, por exemplo, que em certo momento o grupo de pesquisa possa disponibilizar os dados de um escaneamento acessível a alunos, professores e pesquisadores; porém, mantendo a base de informações em um servidor interno.

## **Referências**

Bordin, F.; Teixeira, E.C.; Rolim, S.B.A.; Tognoli, F.M.W.; Veronez, M.R.; Silveira Jr., L.G.; Souza, C.F.N. 2013 (*in press*). Methodology for Acquisition of Intensity Data in Forest Targets using Terrestrial Laser Scanner. IERI Procedia.

Burton, D., Dunlap, D. B., Wood, L. J., and Flaig, P. P., 2011, Lidar intensity as a remote sensor of rock properties: Journal of Sedimentary Research, v. 81, p. 339–347.

Huang, Y.; Peng, J.; Kuo, C.J.; Gopi, M. 2006. Octree-Based Progressive Geometry Coding of Point Clouds. Proceedings of IEEE/Eurographics Symposium on Point-Based Graphics (PBG '06), pp. 103-110.

Inocencio, L.C. 2012. Classificação de Padrões Espectrais em Dados LIDAR para a Identificação de Rochas em Afloramentos, Dissertação de Mestrado, Universidade do Vale do Rio dos Sinos, 35p.

Kovac, B.; Zalik, B. 2010. Visualization of lidar datasets using point- based rendering technique. Computers & Geosciences., 36: 1443–1450.

Pajarola, R. 2005. Stream-processing points. In: Proceedings IEEE Visualization, 10p. Available at <http://www.ifl.uzh.ch/vmml/publications/older-puclications/PStream.pdf>

Souza, M.K.; Veronez, M.R.; Terrestrial Laser Scanning: Application for Measuring of Structures Information in Geological Outcrops, International Journal of Advanced Remote Sensing and GIS 2013, Volume 2, Issue 1, pp. 260-270