



# Método semiautomático para restituição interativa de estradas

Semiautomatic method for interactive roads restitution

*Marcel Emanuelli Rotunno*<sup>1,2</sup>  
*Guilherme Lucio Abelha Mota*<sup>1</sup>

Recebido em julho de 2017.  
Aprovado em abril de 2018.

## RESUMO

Este trabalho apresenta um método semiautomático e interativo de vetorização de estradas. O objetivo é possibilitar um procedimento mais rápido e mais confortável para o operador do que uma ferramenta de restituição manual, onde todos os pontos são medidos pelo operador. A proposta consiste em, valendo-se de imagens de sensoriamento remoto georreferenciadas, agregar de forma automática um par de pontos intermediários a cada segmento fornecido pelo usuário, transformando esse segmento em uma polilinha. A fim de ajustar o vetor à estrada presente na imagem, a abordagem usa um grafo e uma função objetivo que consideram os dados de sensoriamento remoto. A solução da função objetivo fornece o caminho ótimo no grafo, revelando o par de pontos intermediários a ser agregado ao segmento. Experimentos compararam o método proposto à abordagem manual. A base de dados experimental continha 100 recortes com dimensões 4 x 4 km (800 x 800 px) obtidos de imagens orbitais Rapideye com resolução espacial de 5 metros. Estes recortes foram divididos em cinco classes conforme o revestimento da estrada e de seu entorno. O método proposto apresentou dificuldades na classe “estrada pavimentada cercada por solo exposto”. Para as demais classes o resultado produzido apresentou uma redução do tempo de vetorização entre 10% e 21% e uma redução do número de pontos a serem medidos pelo usuário que variou entre 48% e 65%. Em termos do Padrão de Exatidão Cartográfica (PEC), a qualidade dos resultados indicou o uso do método proposto no mapeamento Classe A na escala 1:50.000.

**PALAVRAS-CHAVE:** Extração de Estradas. Base de Dados Vetorial. Cartografia.

---

<sup>1</sup>Universidade do Estado do Rio de Janeiro – UERJ. Programa de Pós-Graduação em Ciências Computacionais – CCOMP, Brasil. E-mail: guimota@ime.uerj.br

<sup>2</sup>Instituto Brasileiro de Geografia e Estatística - IBGE. Coordenação de Cartografia – CCAR, Brasil. E-mail: marcel.rotunno@ibge.gov.br

## **ABSTRACT**

This work presents an interactive semiautomatic method for roads plotting which aims at making roads plotting faster and more comfortable than a manual approach, in which all coordinates are to be measured by the operator. Taking georeferenced remote sensing images into account, the proposed tool adds a couple of intermediate points to each segment given by the user, turning such segment into a polyline. In order to fit such polyline to the respective road present in a remote sensing image, the approach makes use of a directed acyclic graph and an objective function modeled to allow for the remote sensing information. The solution for such a graph optimization problem provides the pair of intermediate points to be aggregated to that segment given by the user. The database employed in the experiments is composed of 100 cutouts with dimensions 4 x 4 km (800 x 800 px) obtained from Rapideye orbital images with spatial resolution of 5 meters. These cutouts were divided into five classes according to the road covering and its surroundings. Method presented difficulties to deal with class “paved road surrounded by bare soil”. For the remaining classes the method presented reduction in the vector extraction time between 10% and 21% and in the number of points to be provided by the user that varied between 48% and 65%. The analysis of the accuracy of the results obtained for such Rapideye images, considering National Cartographic Accuracy Standard of Brazil (PEC), indicated the usefulness of the tool for Class A mapping in the 1: 50,000 scale.

**KEYWORDS:** Road Extraction. Vector Databases. Cartography.

\* \* \*

## **Introdução**

Estradas formam uma classe relevante da cartografia, uma vez que são essenciais no sistema de transporte e servem de referência à localização no espaço geográfico de estabelecimentos comerciais, residenciais e serviços públicos. Em cartas e bases cartográficas, estradas são normalmente representadas através dos respectivos eixos centrais.

Uma possível forma de extração do eixo central de estradas é através do trabalho de campo, utilizando os equipamentos adequados, como rastreadores GNSS (Global Navigation Satellite System), estação total etc. Entretanto, os custos derivados da mobilização de equipes e do aluguel dos equipamentos são muito elevados. Assim, o mais usual é o mapeamento através da vetorização usando como referência imagens aéreas ou orbitais, onde os custos são significativamente menores.

Tradicionalmente, o mapeamento de estradas é feito de forma manual. No passado, empregava-se restituidores analógicos, atualmente, o mapeamento é realizado em ambientes SIG (Sistemas de Informações Geográficas). O método manual fornece resultados precisos, entretanto, é lento e demanda bastante esforço do operador (DA SILVA e CENTENO, 2010), o que gera um desgaste físico e mental que se intensifica ao longo do tempo, aumentando a possibilidade de resultados equivocados. Além disso, os resultados obtidos estão fortemente atrelados à qualidade, ao treinamento, à experiência e ao nível de energia e de atenção do operador. Por essas razões, pesquisas científicas presentes na literatura (ANIL e NATARAJAN, 2010; DAL-POZ et al., 2005; ZHANG, 2006; LONG e ZHAO, 2005; CHENG, 1995; MOKHTARZADE e ZOEJ, 2007; Zhang et al., 1999; SONG; CIVCO, 2004; PANDIT, 2009; BAUMGARTNER et al., 2002; GRUEN e LI, 1997) têm apresentado abordagens automáticas ou semiautomáticas dedicadas à restituição de estradas a partir de imagens digitais. Dentre as intencionalidades motivadoras de tais iniciativas podem ser destacadas a redução do esforço e, conseqüentemente, o aumento do conforto e da produtividade do processo.

Os métodos de extração totalmente automáticos caracterizam-se pela ausência de interferência do operador no processo de restituição (GALLIS et al., 2005), o que limita o trabalho do usuário ao processo de pós-edição. A extração plenamente automática é um ideal a ser alcançado, entretanto, o que se vê na prática é que a maior parte desses métodos acaba gerando resultados imprecisos e incompletos, o que além de tender a deixar o trabalho de pós-edição mais árduo que o próprio método manual, pode prejudicar a qualidade da restituição. Os métodos automáticos, portanto, impõem, ao operador, duas tarefas na etapa de pós-edição: a validação visual de cada uma das feições extraídas e a posterior correção das inconsistências.

Métodos automáticos de extração de estradas normalmente empregam algoritmos de processamento de imagens – com destaque para o

uso de técnicas de segmentação de imagens. Anil e Natarajan (2010) propõem a segmentação através da junção estatística de regiões, Dal-Poz et al. (2005) partem da segmentação de Canny (1986), Zhang (2006) utiliza a segmentação através do algoritmo k-médias, Long e Zhao (2005) usam o deslocamento médio (CHENG, 1995), Mokhtarzade e Zoej (2007) usam uma segmentação através de uma rede neural, Zhang et al. (1999) usam o algoritmo ISODATA (BALL e HALL, 1965).

Outro procedimento bastante utilizado no contexto da extração de estradas é o afinamento (KARATHANASSI, 1999; SHARMA, BEDI e DOGRA, 2013), normalmente empregado para reduzir a largura das regiões de estrada para a ordem de um pixel. Observa-se, por vezes, na literatura, o resultado do afinamento ser submetido a uma vetorização automática.

Os métodos de extração semiautomáticos, por outro lado, conjugam o conhecimento e habilidade do operador com certo grau de automatização da tarefa de extração (DAL POZ e VALE, 2003). Por conseguinte, tendem a apresentar resultados superiores aos produzidos pelos métodos automáticos. Esse fato se deve, em certa medida, à dificuldade intrínseca do reconhecimento de feições cartográficas em imagens, consequência da variedade na geometria e na resposta espectral. Apesar das dificuldades, operadores humanos são capazes de realizar identificação de feições cartográficas com maestria. Assim, em abordagens semiautomáticas, a extração é realizada após o usuário prover alguma informação preliminar ao método.

Métodos semiautomáticos podem usar segmentações supervisionadas para separar as estradas do fundo. Song e Civco (2004) aplicaram como classificador uma SVM (Support Vector Machine) (CORTES e VAPNIK, 1995), Pandit (2009) propõe um algoritmo chamado ATM-R (Adaptive Texture Matching for Road Extraction) no qual o operador fornece alguns retângulos correspondentes a estradas e o algoritmo detecta se há retângulos adjacentes que fazem parte da estrada com base na semelhança de textura.

Em outras abordagens alternativas, o operador provê alguns pontos sementes ou um ponto e uma direção. Esses algoritmos podem ser separados em duas categorias: otimizadores de caminho e rastreadores de caminho (AMO, MARTINEZ e TORRE, 2006).

Dados os pontos inicial e final, otimizadores de caminhos são destinados a achar o traçado da estrada dados os limites fornecidos. Hu et al. (2004) propõem um exemplo de otimizador de caminho em que os operadores fornecem pontos sementes, que descrevem a estrada de forma aproximada, e o algoritmo traça uma parábola para cada par de segmentos que o operador fornece.

Por outro lado, rastreadores de caminho buscam delinear a estrada a partir de um ponto e uma direção inicial. Baumgartner et al. (2002) propõem uma ferramenta rastreadora de caminho na qual o usuário provê um pequeno trecho inicial da estrada e uma seção perpendicular. Usado para calcular com precisão o próximo ponto da estrada, o denominado perfil de referência corresponde à média, ao longo do trecho, das seções paralelas à seção perpendicular.

A abordagem que inspirou a elaboração da presente proposta corresponde ao otimizador de caminho proposto por Gruen e Li (1997). O procedimento adotado recebe como entrada uma polilinha inicial provida pelo usuário e constituída de pontos sementes que representam a estrada de forma grosseira. A abordagem de Gruen e Li aplica a esta polilinha inicial um algoritmo iterativo de inserção e otimização de pontos. O método subjacente se baseia em um modelo conceitual e analítico da estrada. Tal modelo possibilita que a polilinha final, que representa a estrada, seja obtida por um método iterativo baseado em programação dinâmica (BELLMAN, 1954) ou LSB-Snakes.

Por sua vez, a presente proposta estabelece uma abordagem interativa, permitindo que o operador interaja com a construção da polilinha de forma incremental. A intenção é manter controle sobre o foco e a complexidade da decisão pelo local de medição do próximo ponto. Isto se dá

uma vez que a cada etapa o ponto inicial corresponde necessariamente ao último ponto informado pelo usuário (o que é válido a partir do segundo ponto medido pelo usuário) e o candidato a ser indicado como próximo ponto se encontra na coordenada do cursor do mouse. Para facilitar a decisão do operador, uma prévia do traçado da polilinha é apresentada em tempo real ao usuário.

Portanto, o método subjacente a este procedimento, diferentemente do que ocorria em Gruen e Li (1997), recebe como entrada somente os dois pontos que delimitam as extremidades de um segmento. Basicamente, o algoritmo otimizador de caminho proposto inclui dois pontos intermediários ao longo desse segmento. A abordagem utiliza, em linhas gerais, um grafo orientado acíclico para modelar o espaço de busca e obtém o caminho ótimo nesse grafo por programação dinâmica. A solução se baseia em versões adaptadas dos modelos conceituais de estrada e da função objetivo apresentados em (GRUEN; LI, 1997) e produz um ajustamento coerente entre a polilinha obtida e a respectiva estrada presente numa imagem de sensoriamento remoto.

A hipótese que motivou a presente pesquisa considera que a transparência do resultado fornecido pela prévia do traçado em tempo real e a complexidade controlada possibilitada pelo procedimento incremental podem fornecer ganho de produtividade em relação a uma abordagem análoga puramente manual sem comprometer a qualidade da polilinha produzida.

Assim, o objetivo deste artigo é apresentar a proposta, juntamente com a avaliação da qualidade e do ganho de produtividade em relação à vetorização manual. Para tanto, o restante do artigo está organizado da seguinte forma. A seção 2 descreve em detalhe o método proposto, enquanto, a seção 3 descreve e analisa os resultados experimentais. Por fim, na seção 4, são apresentadas as conclusões e sugestões para trabalhos futuros.

## 2 Método

A presente seção se destina à descrição do método proposto e dos modelos conceituais subjacentes à proposta. Primeiramente, são apresentadas a modelagem conceitual da estrada e a modelagem matemática da função objetivo. Em seguida, o algoritmo otimizador de caminho, responsável pela inserção de pontos, incluindo o traçado do caminho ótimo no grafo, é descrito em detalhe. Por fim, são apresentadas considerações a respeito da implementação do software utilizado na prova de conceito e nos experimentos.

### 2.1 Modelo conceitual da estrada

A definição de um modelo conceitual é essencial em um método de extração semiautomática de estradas. Tal descrição deve, sobretudo, expressar a aparência assumida pelas estradas nas imagens. A literatura apresenta com frequência duas hipóteses distintas sobre a aparência de uma estrada em uma imagem em função de sua resolução espacial: (1) elemento linear; (2) área alongada. A hipótese linear é bastante adotada no contexto de imagens de média e baixa resolução, enquanto a hipótese de área alongada se remete a imagens de alta resolução.

No presente trabalho, adota-se o modelo linear, adequado para estradas em áreas rurais cujas larguras sejam representadas por poucos pixels da imagem. O modelo conceitual de estrada toma corpo a partir da aparência geométrica e radiométrica da estrada nas imagens de sensoriamento remoto. No caso específico, o modelo pode ser compreendido como uma adaptação do empregado em (GRUEN e LI, 1997), pois leva em consideração somente parte das propriedades geométricas e radiométricas originais. No presente trabalho, o modelo conceitual considera que:

- a) estradas possuem refletância mais elevada que suas adjacências;

- b) ao longo de uma estrada a variação da refletância é pequena;
- c) a curvatura da estrada deve ser suave ao longo de toda sua extensão.

## 2.2 Modelagem matemática da função objetivo

Esta seção apresenta a formalização matemática do modelo conceitual da estrada aplicado no presente trabalho. O presente modelo possui semelhanças com o empregado em (GRUEN e LI, 1997) análogas ao que ocorre no caso do modelo conceitual da estrada.

Outra adaptação realizada na presente modelagem advém das características espectrais das imagens de entrada. Apesar de ser possível generalizar esta descrição para o caso de múltiplas bandas, nesta apresentação a imagem é tons de cinza, tendo sido formada pela média das bandas R, G e B fornecidas pelo sensor.

Assim sendo, em uma dada imagem,  $\mathbf{a}$  representa um pixel de coordenadas  $(x_a, y_a)$ , enquanto  $f(x_a, y_a)$  corresponderá à sua intensidade, expressa através de um número digital. Doravante, uma polilinha  $\mathbf{P}$  constituída por  $n$  pontos será representada por uma sequência de coordenadas de pixels  $(\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_n)$ , onde um ponto qualquer  $\mathbf{p}_i$  possui coordenadas  $(x_i, y_i)$ .

A primeira propriedade, de forma análoga ao primeiro pressuposto do modelo conceitual da estrada, enuncia que, quando comparados ao entorno da via, os pixels da estrada apresentam refletância superior. Esta propriedade é modelada pela soma dos quadrados dos valores de intensidade dos pixels que compõem a polilinha. A assim chamada  $Prop_1(\mathbf{P})$  é descrita pela Equação 1, onde  $(x_i, y_i)$  são as coordenadas dos vértices que compõem a polilinha  $\mathbf{P}$ , para todo  $i \in \{1, 2, \dots, n\}$ .

A segunda propriedade expressa que a variação da refletância ao longo de uma estrada é suave. A modelagem desta propriedade considera que uma polilinha é formada por um conjunto de segmentos de reta, aqui



denominados fragmentos, e que cada fragmento é delimitado por dois vértices consecutivos da polilinha. Como os fragmentos da polilinha são conectados, em geral, cada vértice pertence a dois fragmentos. As exceções, vértices inicial e final da polilinha, pertencem somente aos fragmentos inicial e final respectivamente.

Para o cálculo da resposta espectral média de cada fragmento, é adotada uma simplificação, tomando a média dos valores dos pixels que limitam esse fragmento. A propriedade  $Prop_2(\mathbf{P})$  computa a soma dos quadrados das diferenças entre o valor de um pixel que é vértice da polilinha e a resposta espectral média do fragmento ao qual esse pixel pertence. Deve ser enfatizado que o fragmento é definido pelos pixels  $\mathbf{p}_i$  e  $\mathbf{p}_{i+1}$ , conforme é retratado pela Equação 2, onde, para cada fragmento, o índice  $j$  assume os valores correspondentes aos limites do fragmento.

A terceira propriedade estabelece que a curvatura da estrada é suave. A modelagem matemática é feita usando os ângulos de deflexão entre os fragmentos que formam a polilinha. A propriedade  $Prop_3(\mathbf{P})$  é descrita pela Equação 3, onde  $\theta_i$  é o ângulo de deflexão entre o fragmento formado pelos pontos  $(x_{i-1}, y_{i-1})$  e  $(x_i, y_i)$  e o fragmento formado pelos pontos  $(x_i, y_i)$  e  $(x_{i+1}, y_{i+1})$ , enquanto  $|\Delta s_i|$  simboliza o comprimento do fragmento formado pelos pontos  $(x_{i-1}, y_{i-1})$  e  $(x_i, y_i)$ . A Figura 1 demonstra o ângulo de deflexão entre esses fragmentos. O ângulo de deflexão é medido a partir do prolongamento do fragmento formado por  $(x_{i-1}, y_{i-1})$  e  $(x_i, y_i)$  em relação ao fragmento formado por  $(x_i, y_i)$  e  $(x_{i+1}, y_{i+1})$ . Esse ângulo varia de zero grau, para fragmentos adjacentes, a 180 graus, no caso de fragmentos coincidentes.

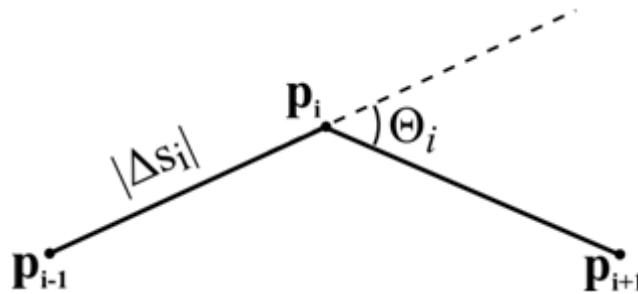
$$Prop_1(\mathbf{P}) = \sum_{i=1}^n [f(x_i, y_i)]^2 \quad (1)$$

$$Prop_2(\mathbf{P}) = \sum_{j=i}^{i+1} \sum_{i=1}^{n-1} \left[ f(x_j, y_j) - \frac{f(x_i, y_i) + f(x_{i+1}, y_{i+1})}{2} \right]^2, \quad (2)$$

$$Prop_3(\mathbf{P}) = \sum_{i=2}^{n-1} \frac{[1 + \cos(\theta_i)]}{|\Delta s_i|}, \quad (3)$$

$$E(\mathbf{P}) = (Prop_1(\mathbf{P}) - Prop_2(\mathbf{P}))Prop_3(\mathbf{P}) \quad (4)$$

Figura 1 - Ângulo de deflexão entre dois fragmentos consecutivos



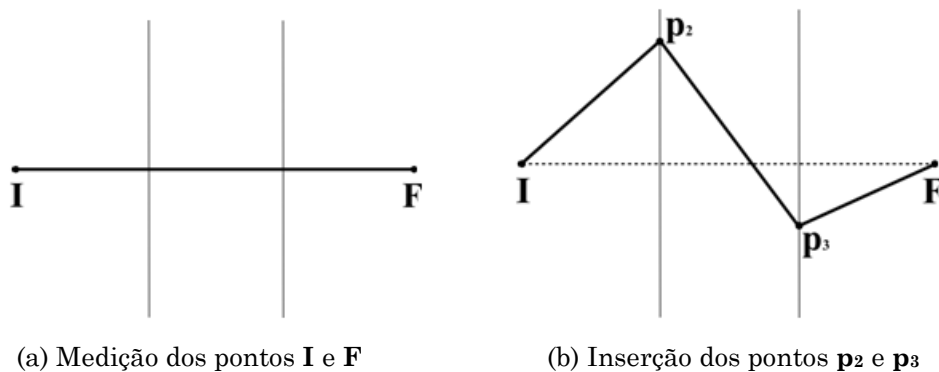
Fonte: elaborada pelo autor

A função objetivo oriunda da modelagem matemática é formada combinando as três propriedades, onde se procura maximizar  $Prop_1(\mathbf{P})$  e  $Prop_3(\mathbf{P})$  enquanto  $Prop_2(\mathbf{P})$  é minimizada. Assim, a função objetivo usada na presente proposta é descrita por  $E(\mathbf{P})$ , conforme apresentado na Equação 4.

### 2.3 Algoritmo de inserção de pontos

O procedimento de inserção de pontos é ilustrado pela Figura 2. Em linhas gerais, o algoritmo proposto recebe como entrada um segmento  $\overline{\mathbf{IF}}$ , conforme representado pela Figura 2(a). Neste contexto,  $\mathbf{I}$  e  $\mathbf{F}$  correspondem a um par de pontos consecutivos fornecidos pelo operador. A partir da definição de  $\overline{\mathbf{IF}}$ , são geradas duas retas perpendiculares ao segmento que conjuntamente o subdividem em três partes idênticas. A solução ótima para as localizações dos pontos  $\mathbf{p}_2$  e  $\mathbf{p}_3$ , inseridos pelo algoritmo, é exemplificada pela Figura 2(b). Cada um dos pontos inseridos estará necessariamente sobre uma das retas perpendiculares a  $\overline{\mathbf{IF}}$ .

Figura 2 – Procedimento de inserção de pontos



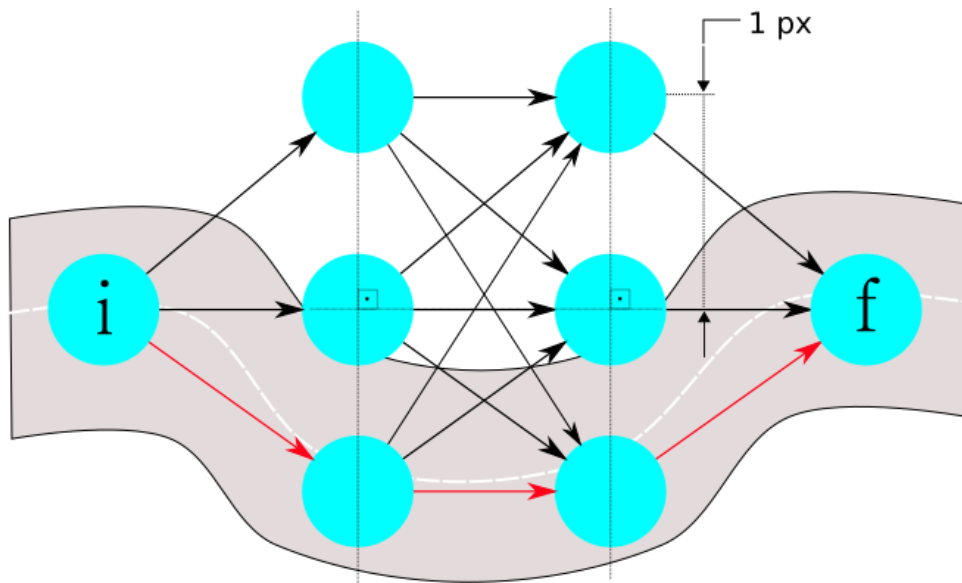
Fonte: elaborada pelo autor

Na presente proposta, dentre as alternativas, a solução ótima é definida através de programação dinâmica (HOROWITZ e SAHNI, 1984). No caso específico, o espaço de busca é definido por um grafo acíclico dirigido com quatro camadas, diferentemente dos grafos com três níveis utilizados em (GRUEN e LI, 1997). Assim, a presente pesquisa demandou a definição de um procedimento de busca pela solução ótima dedicado a grafos com quatro níveis.

O modelo geral para a solução deste problema se ampara num grafo acíclico dirigido que representa todo o espaço de soluções. Especificamente, a busca no grafo visa à obtenção do caminho ótimo entre **I** e **F**, maximizando a função objetivo  $E(\mathbf{P})$ , previamente apresentada na Equação 4.

Um exemplo de grafo acíclico dirigido para este problema é apresentado na Figura 3. Como mencionado anteriormente, o grafo é composto de quatro etapas das quais a primeira engloba apenas o ponto **I**, enquanto a final engloba apenas o ponto **F**. Os vértices referentes a cada etapa intermediária do grafo estão situados sobre as retas perpendiculares ao segmento  $\overline{IF}$  anteriormente apresentadas. É importante enfatizar que os vértices de cada etapa estão conectados a todos os vértices da etapa seguinte.

Figura 3 – Exemplo de grafo definindo o espaço de busca



Fonte: elaborada pelo autor

Um parâmetro que potencialmente impacta o desempenho do método ora em tela corresponde ao espaçamento entre os vértices de cada etapa. A escolha para o valor ótimo para este parâmetro pode ser definida por tentativa e erro, entretanto, na presente implementação, considerou-se, simplesmente, como espaçamento a dimensão do *ground resolution element* (largura do pixel no terreno). A título de ilustração, a Figura 3 apresenta um exemplo de grafo que contém três vértices nas suas etapas intermediárias. O caminho ótimo obtido está marcado em vermelho.

#### 2.4 Implementação do método

O protótipo empregado na avaliação do método foi implementado como um *plugin* para o software QGIS. O doravante chamado *plugin* Adaplin foi concebido como uma ferramenta SIG de criação vetorial com foco na usabilidade. Esta extensão foi desenvolvida na linguagem Python (LAWHEAD, 2015) usando a biblioteca PyQGIS (SHERMAN, 2014) proveniente do QGIS (QGIS Development Team, 2017). Utilizou-se também a biblioteca PyQt (HARWANI, 2011), elaborada na linguagem Python com

base na biblioteca livre Qt (THELIN, 2007), comumente usada para criação de interfaces gráficas.

Durante o uso da ferramenta, à medida que o operador vai extraindo a polilinha, o *plugin* vai armazenando os respectivos pontos em uma lista, que é uma estrutura ordenada e mutável típica da linguagem Python (SUMMERFIELD, 2012).

A partir do segundo ponto fornecido pelo operador, cada vez que um ponto é medido, o grafo do espaço de solução é construído e aplica-se o algoritmo de otimização para a definição do melhor caminho. Em seguida, os pontos obtidos pelo método e o medido pelo operador são incluídos na lista.

A quantidade de vértices adotada para as etapas intermediárias foi de 11, valor estabelecido empiricamente, sendo que um fica localizado no cruzamento entre o segmento e cada reta perpendicular. Dentre os 10 vértices restantes, cinco ficam acima e cinco ficam abaixo do segmento. Foi adotado o espaçamento 4,5 metros entre os vértices de uma etapa, um pouco menor que a resolução espacial das imagens Rapideye usadas nos experimentos do presente trabalho.

Outra prática adotada foi o uso do sistema de referência de uma projeção cartográfica. Assim, as distâncias entre os pontos são dadas em metros ao invés de pixels, como seria se fosse adotado o espaço imagem, ou de graus, caso fosse adotado um sistema de coordenadas geográficas, com coordenadas de longitude e latitude.

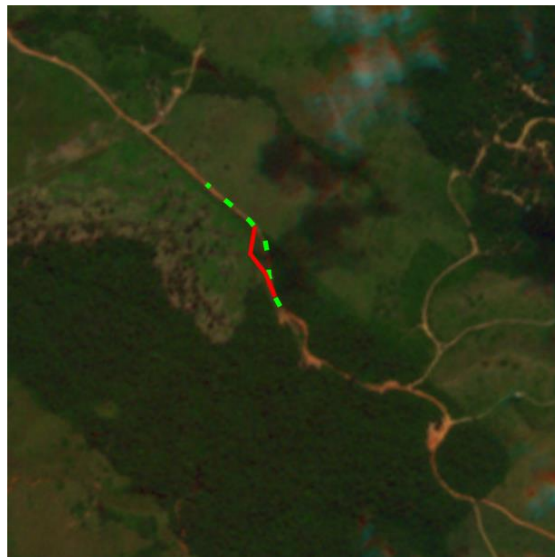
O cálculo das propriedades  $Prop_1(\mathbf{P})$  e  $Prop_2(\mathbf{P})$  requer a verificação dos valores de intensidade dos pixels da imagem de sensoriamento remoto subjacente. Sua obtenção demanda, por sua vez, o acesso aos valores das bandas da imagem para um dado pixel. Para tanto, o *plugin* vale-se de uma função do QGIS que intercepta uma imagem georeferenciada em uma certa posição do espaço objeto.

O procedimento de utilização do *plugin* Adaplin se inicia com a escolha de alguma camada do tipo polilinha, a ser posta em modo de edição. Em seguida, o operador precisará pressionar o botão do Adaplin e eleger a

camada *raster* sobre a qual deseja realizar o traçado vetorial. A partir de então, o operador realizará o traçado vetorial da polilinha criando uma ou mais feições sobre a imagem. Em função da posição do cursor do mouse, o *plugin* gera em tempo real uma previsão do traçado da polilinha para auxiliar o operador na decisão de escolha do ponto final. Completada a edição, o operador deve salvar as alterações, como em uma ferramenta SIG convencional de criação de vetores.

No uso prático da ferramenta, verifica-se, ainda, que, por vezes, o traçado não se adapta à estrada com exatidão. Assim, pode haver, em virtude da existência de sombras ou outras formas de oclusão da estrada, divergências entre o traçado produzido e o observado. Devido à sombra de uma nuvem, este problema se manifesta na imagem apresentada na Figura 4, onde a linha tracejada verde representa o traçado real e a linha vermelha o produzido pelo método.

Figura 4 - Inconsistência produzida pelo método proposto em função de oclusão na estrada



Fonte: elaborada pelo autor

Para estas situações, o operador pode utilizar o modo manual, bastando, para tanto, pressionar a tecla `Ctrl` que o processo de inserção e otimização de pontos passa a estar desabilitado.

### 3 Experimentos

#### 3.1 Parâmetros de análise da qualidade de uma polilinha

Para a avaliação da qualidade da polilinha extraída de forma semiautomática, é necessário compará-la com uma polilinha de referência supostamente isenta de erros. A métrica adotada no presente trabalho é baseada na distância de Hausdorff (ZHAO, SHI e DENG, 2005). Supondo duas polilinhas que percorrem trajetos semelhantes, uma extraída com base em uma ferramenta e outra de referência, a distância de Hausdorff retorna o afastamento máximo entre as duas polilinhas. Sendo **A** e **B** dois conjuntos de pontos que representam o eixo extraído e o de referência, **a** e **b** pontos pertencentes a **A** e **B** respectivamente, a distância de Hausdorff,  $d_H(\mathbf{A}, \mathbf{B})$  pode ser obtida pela Equação 5, onde  $d(\mathbf{a}, \mathbf{b})$  e  $d(\mathbf{b}, \mathbf{a})$  medem a distância euclidiana entre **a** e **b**. No cálculo há a determinação, para cada vértice **a** pertencente a **A**, do ponto **b** pertencente a **B** cuja distância euclidiana para **a** seja mínima. Sendo assim, há um conjunto de distâncias mínimas de **A** para **B**, uma para cada ponto **a** pertencente a **A**. Entre esse conjunto de distâncias, o máximo, conceitualmente representando o maior afastamento local, é retornado. Por outro lado, o mesmo procedimento é aplicado para as distâncias de **B** para **A**. Deste modo, há dois maiores afastamentos, um de **A** para **B** e outro de **B** para **A**, e o máximo entre esses equivale à distância de Hausdorff.

Como a distância de Hausdorff retorna o afastamento máximo entre duas polilinhas, seu valor não representa tão bem a similaridade entre elas, pois elas podem ser coincidentes em grande parte de sua extensão e em um pequeno trecho apresentar um desvio grande. Para resolver essa questão, Calvache (2007) propõe uma extensão da distância de Hausdorff, tomando, ao invés do máximo, a média e o desvio padrão dos conjuntos de distâncias

mínimas. A média dos afastamentos é representada por  $\overline{d_H}$  (A, B) na Equação 6.

Por sua vez, o desvio padrão dos afastamentos é representado por  $\sigma_H$  na Equação 7.

$$d_H(\mathbf{A}, \mathbf{B}) = \max\left\{\max_{\mathbf{a} \in \mathbf{A}} \left\{\min_{\mathbf{b} \in \mathbf{B}} d(\mathbf{a}, \mathbf{b})\right\}, \max_{\mathbf{b} \in \mathbf{B}} \left\{\min_{\mathbf{a} \in \mathbf{A}} d(\mathbf{b}, \mathbf{a})\right\}\right\}, \quad (5)$$

$$\overline{d_H}(\mathbf{A}, \mathbf{B}) = \max\left\{\text{media}_{\mathbf{a} \in \mathbf{A}} \left\{\min_{\mathbf{b} \in \mathbf{B}} d(\mathbf{a}, \mathbf{b})\right\}, \text{media}_{\mathbf{b} \in \mathbf{B}} \left\{\min_{\mathbf{a} \in \mathbf{A}} d(\mathbf{b}, \mathbf{a})\right\}\right\} \quad (6)$$

$$\sigma_H(\mathbf{A}, \mathbf{B}) = \max\left\{\sigma_{\mathbf{a} \in \mathbf{A}} \left\{\min_{\mathbf{b} \in \mathbf{B}} d(\mathbf{a}, \mathbf{b})\right\}, \sigma_{\mathbf{b} \in \mathbf{B}} \left\{\min_{\mathbf{a} \in \mathbf{A}} d(\mathbf{b}, \mathbf{a})\right\}\right\} \quad (7)$$

Os parâmetros  $d_H$ ,  $\overline{d_H}$  e  $\sigma_H$  são usados como parâmetros na avaliação da qualidade do eixo extraído. Um *script* Python no software QGIS foi usado para o cálculo desses parâmetros. Antes da realização do cálculo, esse *script* substitui os pontos inicial e final do eixo extraído pelos pontos inicial e final do eixo de referência, igualando os extremos da trajetória das duas polilinhas, o que evita a superestimação das medidas de afastamento das polilinhas. Em seguida, o *script* realiza o adensamento da quantidade de pontos nas polilinhas através de uma interpolação linear com o intervalo de um metro, o que faz com que a forma de uma polilinha seja mais fielmente representada que simplesmente pelo conjunto de pontos que a constitui.

### 3.2 Base de dados

A base de dados usada nos experimentos foi composta por recortes de imagens Rapideye. A constelação Rapideye é formada por cinco satélites equipados com sensores idênticos, os quais captam imagens em cinco bandas: azul, verde, vermelho, vermelho limítrofe e infravermelho próximo. A resolução espacial dessas imagens é de 5 metros e o nível de correção das imagens adquiridas é 3A (RapidEye Ortho Product), onde são aplicadas a



ortorretificação, correções radiométrica, geométrica e de terreno em uma projeção cartográfica (Planet Labs Team, 2016). A resolução radiométrica das imagens distribuídas é de 16 bits.

Os recortes extraídos das imagens buscaram abranger certas estradas e possuíram dimensões padronizadas de 4 x 4 km, o que equivale a 800 x 800 px na imagem. Para uma melhor análise dos resultados obtidos através do método semiautomático, os recortes foram divididos em classes conforme o revestimento da estrada e a cobertura do solo em seu entorno. O número de classes usado foi de cinco, com 20 recortes para cada classe, o que totaliza 100 recortes. O nome de cada classe e a respectiva descrição são apresentados na **Tabela 1**.

Tabela 1 – Nome e descrição das classes de estradas

<b>Classe</b>	<b>Nome completo</b>	<b>Descrição</b>
<b>PAV-RAS</b>	Estrada pavimentada cercada por vegetação rasteira	Estradas de asfalto ou concreto cercadas por vegetação pouco desenvolvida
<b>PAV-FLO</b>	Estrada pavimentada cercada por vegetação rasteira e floresta	Estradas de asfalto ou concreto cercadas por áreas com grande densidade de árvores intercaladas por regiões de vegetação rasteira
<b>PAV-SOLO</b>	Estrada pavimentada cercada por solo exposto	Estradas de asfalto ou concreto cercadas por solo sem vegetação
<b>NAT-RAS</b>	Estrada em leito natural cercada por vegetação rasteira	Estradas de terra cercadas por vegetação pouco desenvolvida
<b>NAT-FLO</b>	Estrada em leito natural cercada por vegetação rasteira e floresta	Estradas de terra cercadas por áreas com grande densidade de árvores

Fonte: elaborada pelo autor

Para que seja possível comparar os resultados da extração manual com a realizada de maneira semiautomática, é necessária a geração de dados de referência. Para isso, o eixo central de cada estrada presente nos recortes foi delineado manualmente de forma cuidadosa, empregando o nível máximo de zoom que possibilitasse a estimativa precisa do eixo central da

estrada. Esses eixos de referência foram considerados como sendo isentos de erros.

### 3.3 Procedimento Experimental

Algumas práticas foram empregadas para assegurar a confiabilidade das medições realizadas nos experimentos, dentre as quais destacam-se:

- a) verificação da ergonomia das instalações e dos equipamentos;
- b) o operador foi orientado a produzir enquanto estivesse confortável para que os resultados obtidos não sofressem influência do cansaço;
- c) a vetorização manual foi completada antes de iniciar a semiautomática, evitando a confusão no modo de operação das ferramentas;
- d) duas imagens de treino foram providenciadas para que o operador pudesse exercitar o uso da ferramenta antes de começar a vetorização;
- e) em parte dos experimentos que avaliaram o método semiautomático proposto, o modo manual foi desabilitado para que estes resultados refletissem a dificuldade de lidar com cenários adversos;
- f) o arquivo editado pelo operador ficou armazenado em um banco de dados com acesso multiusuário de forma que o trabalho pudesse ser acompanhado.

A avaliação dos resultados alcançados procurou comparar a produtividade da vetorização, o esforço do operador e a qualidade do vetor extraído. Para a avaliação da qualidade do vetor, foram usados os parâmetros baseados na distância de Hausdorff descritos na seção 3.1. A produtividade foi medida em termos da duração ( $t$ ) do procedimento de extração da estrada presente em cada recorte. O esforço foi uma métrica definida como o número de pontos ( $n_p$ ) fornecidos pelo operador. Para a vetorização manual, o esforço é igual ao número de pontos presentes na polilinha. Para o caso da vetorização com o método ora avaliado, é necessário

considerar a inserção de dois pontos adicionais a cada segmento marcado pelo operador. Portanto, neste caso, sendo  $n$  o número de pontos presentes na polilinha e  $n_p$  o esforço do operador, expresso em termos do número de pontos medidos, é possível deduzir que  $n_p = (n + 2)/3$ .

### 3.4 Resultados e Análise

Nas seções a seguir são apresentados e analisados os resultados experimentais. Na primeira, para todas as imagens de teste, são comparados o método manual e o método proposto permanecendo o método manual desabilitado. Em seguida, para um subconjunto contendo 20 das imagens de teste (quatro imagens por classe), ambas as alternativas utilizadas no experimento anterior foram comparadas ao método semiautomático com possibilidade de uso do método manual nas regiões em que o operador julgar que este pode prover resultados superiores. A partir de agora a possibilidade de intercalar o uso destas duas alternativas é referida pelo termo método híbrido.

#### 3.4.1 Comparação com o método manual

Neste experimento, cada imagem do conjunto de treinamento foi submetida aos métodos de extração manual e semiautomática. Para ambos os métodos, foram medidos os parâmetros de qualidade da polilinha obtida em relação à de referência ( $d_H$ ,  $\overline{d_H}$ ,  $\sigma_H$ ) além de  $t$  (duração) e  $n_p$  (esforço). Para cada recorte, as diferenças entre os valores de cada parâmetro, representadas pelo símbolo  $\Delta$ , foram computadas a fim de estabelecer uma comparação entre os distintos modos de extração. Valores positivos significam um melhor desempenho da extração semiautomática, enquanto valores negativos denotam um melhor desempenho da extração manual. Os valores médios das diferenças obtidas em cada classe de recorte, ora

representados pelo símbolo  $m$ , foram utilizados para sintetizar os resultados. Estes valores são apresentados na **Tabela 2**.

Tabela 2 - Diferenças entre as médias de cada parâmetro

Classe	$m_{\Delta d_H}$ (m)	$m_{\Delta \overline{d_H}}$ (m)	$m_{\Delta \sigma_H}$ (m)	$m_{\Delta t}$ (s)	$m_{\Delta np}$ (#)
PAV-RAS	-6,7	-1	-1,1	8,2	16,3
PAV-FLO	-8,9	-1,4	-1,7	5,5	33,9
PAV-SOLO	-11,2	-4,2	-2,8	-3,1	5,9
NAT-RAS	-7,1	-1,2	-1,2	5,9	32,1
NAT-FLO	-7,3	-1	-1	4,9	39,7

Fonte: elaborada pelo autor

Os resultados da **Tabela 2** mostram, em geral, uma melhor exatidão da polilinha produzida pela extração manual, expressa pelos resultados favoráveis em termos de distância de Hausdorff, materializados nos valores negativos nas três primeiras colunas da tabela. Em contraste, a tabela reflete a melhoria na produtividade possibilitada pelo método semiautomático, com resultados favoráveis em termos de duração e esforço, expressos pelos valores positivos nas duas últimas colunas da tabela. Como exceção observa-se os resultados para a classe PAV-SOLO, que apresentou o pior desempenho para a vetorização semiautomática, com quase todos os resultados desfavoráveis, exceto do parâmetro de esforço.

É importante enfatizar que, embora desfavoráveis ao método semiautomático, as médias das diferenças da distância de Hausdorff média ficam dentro de um pixel da imagem Rapideye, cuja resolução espacial é de cinco metros, o que indica uma boa qualidade do vetor gerado pela ferramenta desenvolvida.

Outra comparação foi realizada em termos de número de vitórias de cada método testado, que expressa o número de recortes em que cada método é superior. Em termos dos parâmetros de exatidão da polilinha, o método manual apresentou resultados superiores para a grande maioria das imagens de teste. Por outro lado, o método semiautomático foi superior em termos de duração e de esforço, requisitando, em geral, menos tempo e menos esforço para a obtenção da polilinha, exceto para a classe PAV-SOLO.

O desempenho apresentado corrobora os resultados apresentados anteriormente pela **Tabela 2**.

### 3.4.2 Comparação dos métodos manual e semiautomático com o método híbrido

Um outro experimento foi realizado para vislumbrar a possível melhoria na exatidão da polilinha através da aplicação do método híbrido, no qual o operador pode alternar entre o modo semiautomático e o manual, quando este julgar necessário.

Com esse intuito, foram escolhidas aleatoriamente, para cada classe, quatro imagens. Além da vetorização através dos métodos manual e semiautomático, a vetorização através do método híbrido também foi realizada.

A **Tabela 3** ilustra os resultados da média expressa em metro dos parâmetros de qualidade para cada classe. Em cada célula, o valor para cada método está separado por uma barra. A ordem de apresentação é: método manual, método híbrido, método semiautomático.

Tabela 3 - Comparação com os resultados fornecidos pelo método híbrido (na ordem: manual / híbrido / semiautomático)

Classe	$m_{d_H}$ (m)	$m_{d_{\bar{H}}}$ (m)	$m_{\sigma_H}$ (m)
PAV-RAS	8,4 / 8,5 / 11,2	2,1 / 2,1 / 2,5	1,6 / 1,8 / 2,1
PAV-FLO	7,0 / 10,6 / 14,2	1,9 / 2,5 / 2,7	1,4 / 2,0 / 2,5
PAV-SOLO	8,3 / 14,2 / 19,7	2,6 / 2,8 / 5,6	1,9 / 2,6 / 4,6
NAT-RAS	6,0 / 9,3 / 12,1	1,9 / 2,1 / 2,4	1,4 / 1,8 / 2,0
NAT-FLO	6,0 / 10,3 / 14,0	1,6 / 2,1 / 2,3	1,2 / 1,7 / 2,3

Fonte: elaborada pelo autor

Pelos resultados obtidos, é possível notar melhores resultados na extração manual. Entretanto, comparando o método híbrido com o semiautomático, percebe-se uma melhora das médias dos parâmetros de qualidade, o que sugere o uso do método híbrido pode possibilitar resultados

com exatidão próxima ao do método manual e, simultaneamente, duração e esforço próximos aos produzidos com o uso do método semiautomático.

Dentre as melhoras obtidas nos distintos parâmetros, a mais perceptível foi a em  $m_{dH}$ . Acredita-se que essa melhora se deve ao fato do operador possuir um maior controle sobre a ferramenta, evitando assim erros acentuados. Nesse sentido, o impacto do uso do método híbrido é mais perceptível na classe PAV-SOLO, já que o solo tende a atrair os pontos inseridos pela vetorização semiautomática.

#### 4 Conclusão

No presente artigo, foi apresentado um método semiautomático e iterativo de vetorização, com o objetivo de otimizar a vetorização de estradas. A ferramenta foi implementada em um ambiente de Sistema de Informações Geográficas, como uma extensão para o software livre QGIS. Os resultados obtidos pelo método delineado revelam um bom potencial, apresentando uma alternativa à vetorização manual.

De forma geral, os resultados evidenciaram uma maior rapidez e conforto com o uso da vetorização semiautomática, enquanto a vetorização manual se mostrou um pouco mais precisa. A classe estrada pavimentada cercada de solo exposto (PAV-SOLO) apresentou os piores resultados para a extração semiautomática, tanto em termos de qualidade, quanto em termos de usabilidade. Nessa classe, o método semiautomático tende a atrair os pontos inseridos para o solo exposto ao invés do asfalto, o que se deve muito provavelmente à semelhança da média das componentes R, G e B da imagem para as classes estrada e solo exposto.

A ferramenta possui limitações para lidar com objetos claros nas margens da estrada, tais como casas, e obstruções na estrada, como as causadas por sombras e árvores. Para superar tais adversidades o modo híbrido foi implementado, então o usuário pode vetorizar manualmente caso a ferramenta esteja apresentando resultados equivocados. Experimentos

realizados indicam uma melhora dos parâmetros de qualidade com o uso do método híbrido, principalmente de  $d_H$ .

Uma análise foi realizada para determinar qual a classe e a escala da carta que seria possível alcançar com base em imagens Rapideye utilizando o método semiautomático, segundo o Padrão de Exatidão Cartográfica (PEC), instituído pelo Decreto Lei número 89.817 de 1984. De acordo com o decreto, uma carta em uma dada escala é classificada quanto à sua exatidão em A, B e C, e deve satisfazer o PEC e o desvio padrão correspondente dessa escala. Para classificar a carta, 90% dos pontos testados no terreno não podem apresentar erro superior ao PEC e o desvio padrão desse conjunto de erros não deve ultrapassar 60,8% do PEC. O PEC planimétrico classe A corresponde a 0,5 mm na escala da carta, sendo de 0,3 mm na escala da carta o desvio padrão correspondente.

É conveniente notar que o decreto trata de feições pontuais. Para aplicá-lo a feições lineares, o desvio máximo  $d_H$  foi tomado como erro linear para que se avalie o pior caso. Os resultados do parâmetro  $d_H$  para os 100 recortes foram agregados. Dentre este conjunto, apenas dois recortes ultrapassaram o erro de 25 metros e o desvio padrão computado foi de 4,8 metros. Tais resultados sugerem ser possível, utilizando imagens Rapideye, a aplicação da ferramenta para o mapeamento de uma carta classe A em escala 1:50.000, cujo valor do PEC é de 25 metros e o desvio padrão do conjunto de pontos de teste deve ser menor ou igual a 15 metros. Contudo, em casos de um cenário semelhante ao da classe PAV-SOLO, ressalva-se que a extração utilizando as facilidades manuais do procedimento híbrido é mais indicada.

A ferramenta desenvolvida atualmente insere dois pontos para cada segmento fornecido pelo usuário. Um passo futuro é o desenvolvimento da inserção de uma quantidade variável de pontos em função do comprimento do segmento. Outro passo é a pesquisa de uma forma de extração do eixo central, pois a ferramenta foi projetada para estradas que se assemelham a elementos lineares, o que acontece quando elas possuem poucos pixels de

largura. Outros critérios de escolha do traçado também devem ser estudados, como a atribuição de um peso maior ao gradiente e o uso da distância euclidiana entre os valores de pixels ao invés da diferença entre médias das componentes R, G e B.

## **Agradecimentos**

Os autores são gratos ao Programa de Pós-Graduação em Ciências Computacionais da Uerj por abrigar a realização da presente pesquisa. Agradecemos também ao Instituto Brasileiro de Geografia e Estatística pela disponibilização das imagens e de um operador que auxiliou na parte experimental, contribuições sem as quais não teria sido possível o desenvolvimento deste trabalho.

## **Referências**

- AMO, M.; MARTINEZ, F.; TORRE, M.. Road extraction from aerial images using a region competition algorithm. **IEEE transactions on image processing**, [s.l.], v. 15, n. 5, 2006 pp. 1192-1201.
- ANIL, PN; NATARAJAN, S. Automatic road extraction from high resolution imagery based on statistical region merging and skeletonization. **International Journal of Engineering Science and Technology**, [s.l.], v. 2, n. 3, 2010. pp. 165-171.
- BALL, Georey H.; HALL, David J. **ISODATA, a novel method of data analysis and pattern classification**, California, Stanford Research Institute, abril 1965.
- BAUMGARTNER, Albert; HINZ, Stefan; WIEDEMANN, Christian. Efficient methods and interfaces for road tracking. **International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences**, [s.l.], v. 34, n. 3/B, 2002. pp. 28-31.
- BELLMAN, Richard. The theory of dynamic programming. **Bulletin of the American Mathematical Society**, [s.l.], v. 60, n. 6, 1954. pp. 503-516.



- CALVACHE, Antonio Tomás Mozas. **Control de calidad posicional en cartografía por elementos lineales**. Tese de Doutorado. Universidad de Jaén, Andalucía (Espanha). 2007. 249 p.
- CANNY, J. A computational approach to edge detection. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 8, n. 6, 1986. pp. 679-698.
- CHENG, Yizong. Mean shift, mode seeking, and clustering. **IEEE Transactions on Pattern Analysis And Machine Intelligence**, [s.I.], v. 17, n. 8, 1995. pp. 790-799.
- CORTES, Corinna; VAPNIK, Vladimir. Support-vector networks. **Machine Learning**, [s.I.], v. 20, n. 3, 1995. pp. 273-297.
- DA SILVA, Claudionor Ribeiro; CENTENO, Jorge Antônio Silva. Extração automática de estradas vicinais com base na Transformada Radon. **Revista Brasileira de Cartografia**, n. 62, 2010. pp. 441–452.
- DAL-POZ, Aluir P.; VALE, Giovane M. do. Metodologia modificada de programação dinâmica para a extração acurada do eixo de rodovia em imagens digitais. **Revista Brasileira de Cartografia**, [s.I.], v.1, n. 55, 2003. pp. 11-20.
- DAL-POZ, Aluir P.; VALE, Giovane M. do; ZANIN, Rodrigo B. Automatic extraction of road seeds from high-resolution aerial images. **Anais da Academia Brasileira de Ciências**, [s.l.], v. 77, n. 3, 2005. pp.509-520.
- GALLIS, Rodrigo Bezerra de Araújo et al. EXTRAÇÃO SEMI-AUTOMÁTICA DE RODOVIAS ATRAVÉS DE OTIMIZAÇÃO POR PROGRAMAÇÃO DINÂMICA NO ESPAÇO OBJETO: CASO MONO. **Revista Brasileira de Cartografia**, v. 3, n. 57, 2005. pp. 168-179.
- GRUEN, Armin; LI, Haihong. Semi-automatic linear feature extraction by dynamic programming and lsb-snakes. **Photogrammetric engineering and remote sensing**, [s.I.], v. 63, n. 8, 1997. pp. 985-994.
- HARWANI, B. M. Introduction to Python Programming and Developing GUI Applications with PyQt. Boston, Estados Unidos: Course Technology Press, 2011.
- HOROWITZ, Ellis; SAHNI, Sartaj. **Fundamentals of computer algorithms**. [s.I.]: Galgotia Publications, 1984.

- HU, Xiangyun; ZHANG, Zuxun; TAO, C. Vincent. A robust method for semiautomatic extraction of road centerlines using a piecewise parabolic model and least square template matching. **Photogrammetric Engineering & Remote Sensing**, [s.I.], v. 70, n. 12, 2004. pp. 1393-1398.
- KARATHANASSI, V.. A thinning-based method for recognizing and extracting peri-urban road networks from SPOT panchromatic images. **International Journal of Remote Sensing**, [s.I.], v. 20, n. 1, 1999. pp.153-168.
- LAWHEAD, Joel. **QGIS Python Programming Cookbook**, Birmingham, Inglaterra: Packt Publishing Ltd., 2015.
- LONG, Hui; ZHAO, Zhongming. Urban road extraction from high-resolution optical satellite images. **International Journal of Remote Sensing**, [s.I.], v. 26, n. 22, 2005. pp. 4907-4921.
- MOKHTARZADE, M.; ZOEJ, M.J. Valadan. Road detection from high-resolution satellite images using artificial neural networks. **International Journal of Applied Earth Observation and Geoinformation**. [s.I.], v. 9, n. 1, 2007. pp. 32-40.
- PANDIT, Vinay. **Automatic road extraction from high resolution satellite imagery**. 2009. 55 f. Dissertação de Mestrado. International Institute of Information Technology, Hyderabad (Índia).
- Planet Labs Team. **Rapideye imagery product specifications**. San Francisco, California, 2016, 50 p. Site: <<https://www.planet.com/products/satellite-imagery/files/160625-RapidEye%20Image-Product-Specifications.pdf>>.
- QGIS Development Team. **QGIS User Guide**, [s.I.], QGIS Project, 2017.
- SHARMA, Nisha; BEDI, Rajeev; DOGRA, Ajay Kumar. A SURVEY ON ROAD EXTRACTION FROM COLOR IMAGE USING VECTORIZATION. **International Journal of Research in Engineering and Technology**, [s.I.], v. 02, n. 10, 2013. pp. 453-458.
- SHERMAN, Gary. **The PyQGIS Programmer's Guide**. Alasca, Estados Unidos: Locate Press, 2014.
- SONG, Mingjun; CIVCO, Daniel. Road extraction using svm and image segmentation. **Photogrammetric Engineering & Remote Sensing**, [s.I.], v. 70, n. 12, 2004. pp. 1365-1371.

- SUMMERFILED, Mark. **Programação em Python 3: uma introdução completa à linguagem Python**, Rio de Janeiro, Brasil: Alta Books Editora, 2012..
- THELIN, Johan. **Foundations of Qt Development**. Nova Iorque, Estados Unidos: Apress, 2007.
- ZHANG, Chunsun; MURAI, Shunji; BALTSAVIAS, Emmanuel. Road network detection by mathematical morphology. In: PROCEEDINGS OF ISPRS WORKSHOP 3D GEOSPATIAL DATA PRODUCTION: MEETING APPLICATION REQUIREMENTS, [s.l.], 1999, Paris, França, pp. 185-200.
- ZHANG, Qiaoping. **Automated Road Network Extraction from High Spatial Resolution Multi-Spectral Imagery**. Tese de Doutorado. University of Calgary, Alberta (Canadá). 2006. 156 p.
- ZHAO, Chunjiang; SHI, Wenkang; DENG, Yong. A new Hausdorff distance for image matching. **Pattern Recognition Letters**, [s.l.], v. 26, n. 5, 2005. pp. 581-586.