

Experimentando Processar e Visualizar Dados de Radares Meteorológicos Usando NumPy e Pygame

Tiago Martinuzzi Buriol

Instituto Federal do Paraná - IFPR

82530-230, Curitiba, PR

E-mail: tiago.buriol@ifpr.edu.br

César Beneti

Instituto Tecnológico Simepar

81531-980, Curitiba, PR

E-mail: beneti@simepar.br

Resumo: A visualização de dados físicos é uma demanda mundial, crescente em diversos setores, devido à grande capacidade que se tem hoje para obtenção de dados a partir de sensores e simulações computacionais. Ferramentas computacionais que permitem interagir e explorar visualmente grandes conjuntos de dados são importantes recursos para a análise e compreensão de fenômenos diversos. No entanto, muitos aspectos, incluindo fatores humanos e computacionais, apresentam desafios consideráveis para o desenvolvimento de ferramentas de visualização eficientes. Neste trabalho é apresentado o desenvolvimento de um visualizador de dados de radar meteorológico, focado na facilidade de uso e de interação. Tem sido utilizada a linguagem Python, juntamente com os pacotes Numpy e Pygame, os quais permitem obter um equilíbrio entre produtividade e desempenho do aplicativo. As estratégias utilizadas para a implementação de algoritmos de processamento numérico e de visualização científica, bem como as vantagens e limitações encontradas são discutidas.

1. INTRODUÇÃO

Atividades científicas e profissionais cada vez mais fazem uso de grandes quantidades de dados gerados diariamente a partir de instrumentos de medição e simulação computacional. As ferramentas computacionais de visualização científica tornaram-se parte indispensável em muitos campos, especialmente em domínios que fazem amplo uso da matemática e de processamento numérico. A visualização provê uma forma eficiente de analisar dados e, então, dar apoio na tomada de decisões. A exploração visual de dados oferece meios de se obter um suporte cognitivo, facilitando a percepção e a construção mental da estrutura destes dados e, conseqüentemente, a criação de um modelo conceitual do fenômeno em estudo.

Embora sejam bem conhecidos os benefícios providos pela representação visual, a visualização e manipulação de dados, a fim de se obter informações pertinentes, de forma eficaz e eficiente, ainda é um desafio. Segundo Wen Qi [5], os principais gargalos são o processamento em tempo real, a visualização de grandes quantidades de dados e a capacidade humana de compreender e interagir com estes dados [1], [3]. O volume e complexidade dos dados a serem analisados extrapola o controle que o observador tem sobre a representação destes dados. Analistas precisam de ferramentas simples e eficazes para representar os dados e para interagir com esta representação a fim de otimizá-la, criando e verificando hipótese.

Nesse sentido, a usabilidade de um sistema de visualização é um aspecto relevante, pois ao inspecionar visualmente um conjunto de dados complexos, é importante que o analista possa se concentrar na análise e na compreensão dos dados e do fenômeno ao invés de preocupar-se em como utilizar o sistema. Isso significa não distrair-se procurando uma funcionalidade em menus e submenus ou realizando inúmeros cliques com o mouse. Uma interface com usuário mais eficiente pode permitir ao analista concentrar-se na investigação dos dados para obter uma melhor compreensão do fenômeno e, assim, conseguir tomar decisões e chegar a conclusões

mais rapidamente. Também se espera que o sistema responda rapidamente às solicitações do usuário, então, o desempenho dos algoritmos numéricos e de visualização deve ser aceitável.

Neste trabalho, é apresentado o uma experiência com o desenvolvimento de uma ferramenta de visualização de dados de radar meteorológico com interface projetada para ser simples e intuitiva, permitindo ao analisa explorar os dados interagindo diretamente com eles. Para isso, foram desenvolvidas ferramentas com controle gráfico que executam processamento otimizado sobre os dados para apresentar a visualização desejada.

Os dados obtidos por radares meteorológicos são distribuídos espacialmente em uma grade estruturada, descrita por meio de um sistema de coordenadas esféricas. Ao gerar imagens para visualização destes dados, uma transformação do sistema de coordenadas esféricas 3D para um sistema de coordenadas cartesianas precisa ser processada. Para obter um bom nível de interação, esse processamento numérico precisa ser rápido, de maneira que o usuário possa, por exemplo, traçar uma linha imediatamente ver o interior do volume de dados em um plano de corte.

Buscando um equilíbrio entre esforço demandado para o desenvolvimento e o desempenho do aplicativo, optou-se por utilizar a linguagem Python, juntamente com os pacotes Numpy e Pygame. Esta linguagem sido indicada como uma boa alternativa para aplicações científicas e desenvolvimento de pesquisas [4]. Utilizando o pacote Numpy, é possível escrever algoritmos para processamento numérico com relativa facilidade e com desempenho satisfatório para nosso propósito. Pygame proporciona uma maneira facilitada de desenvolver sistemas visuais interativos, tais como ferramentas gráficas e jogos de computador [2].

A estratégia adotada foi manter um conjunto de dados na memória da máquina e obter produtos diversos a partir destes dados, interagindo diretamente em uma representação visual que serve como base. Dessa maneira, podemos dar zoom ou obter cortes verticais do volume de dados criando novas imagens a partir dos dados originais, sem haver a necessidade de converter previamente a grade polar em uma grade cartesiana. Por exemplo, podemos obter um corte vertical desenhando uma linha (clique, arrastar e soltar) com o mouse. Ou então, ampliar e transladar gradualmente a imagem utilizando a rodinha do mouse, assim que o movimento cessa a imagem é atualizada a partir dos dados originais.

2. VISUALIZAÇÃO DE DADOS DE RADARES METEOROLÓGICOS

Na última década, a tecnologia de radares meteorológicos tem evoluído e tem estado disponíveis para vários centros operacionais meteorológicas em todo o mundo. As informações disponíveis geradas com esses equipamentos têm, em muitos casos, aumentado mais de cinco vezes. Dessa forma, selecionar, analisar e extrair rápida e facilmente informações relevantes a partir deste conjunto de dados, complexos e multidimensionais, é um desafio.

A visualização, nesse contexto, é uma importante ferramenta, pois provê uma forma eficiente de analisar dados e, então, dar apoio na tomada de decisões. No entanto, para que a visualização ofereça um ganho real na compreensão dos dados e do fenômeno, a ferramenta utilizada deve ser efetiva em questões de usabilidade e desempenho, além de oferecer as funcionalidades certas. Basicamente é desejável que o analista possa obter rápida e facilmente as informações que procura explorando visualmente e interativamente o conjunto de dados.

Tradicionalmente, os sistemas para visualização de dados meteorológicos têm mantido alguns padrões de interface com usuário e de funcionalidades, os quais têm persistido por muitos anos. Os projetistas e programadores desses sistemas têm dado pouca importância para questões relativas à interface, especialmente com relação à usabilidade e à concepção de novas formas de explorar visualmente os dados.

Radars meteorológicos geram dados sobre chuva a partir da emissão de feixes de ondas eletromagnéticas emitidos horizontalmente (varredura azimutal) a cada 1°, completando os 360° e em diferentes elevações da antena, ou seja, diferentes inclinações verticais. O volume de varredura utilizado neste trabalho corresponde a uma sequência de 360 varreduras azimutais com 14 diferentes elevações da antena e 800 pontos igualmente espaçados ao longo de um feixe de 200 km. Por fim, tem-se uma grade em coordenadas polares com resolução de 360x14x800, completando 4.032.000 pontos. Cada um destes pontos podem conter dezenas de variáveis associadas a ele. A cada 10 minutos é gerado um volume de dados.

A partir desses dados são obtido “produtos”, que são imagens 2D representado uma ou mais variáveis, seja em uma projeção de uma das elevações ou um corte no volume. Dentre as dezenas de variáveis que um radar pode medir, destaca-se a refletividade (Z) que é um fator entre a irradiância emitida pelo radar e a recebida por ele depois de retroespalhadas pelos hidrometeoros (gotas de chuva) presentes na atmosfera. A unidade utilizada é o dBZ, que é uma escala logarítmica da refletividade. Os valores variam entre -24 dBZ zero e 63 dBZ e quanto maior forem esses valores, maior serão os diâmetros das gotas presentes no volume medido e, conseqüentemente, maior será a intensidade de precipitação.

Os diversos produtos gerados a partir dos dados de radar precisam ser visualizados e analisados a fim de se obter informações sobre previsão a curto prazo, como possibilidade de tempestades e tormentas. A Figura 1 mostra um sistema 2D para visualização de dados do radar e, também, um volume que consiste na estrutura 3D dos dados como eles são originalmente adquiridos. Portanto, a visualização de dados de radar envolve uma transformação de variáveis distribuídas em um volume descrito em coordenadas esféricas para um sistema de coordenadas cartesianas. Considerando a quantidade de dados e a necessidade de se prover um nível adequado de interatividade, processar a conversão do sistema esférico para cartesiano, com desempenho adequado, não é uma tarefa trivial.

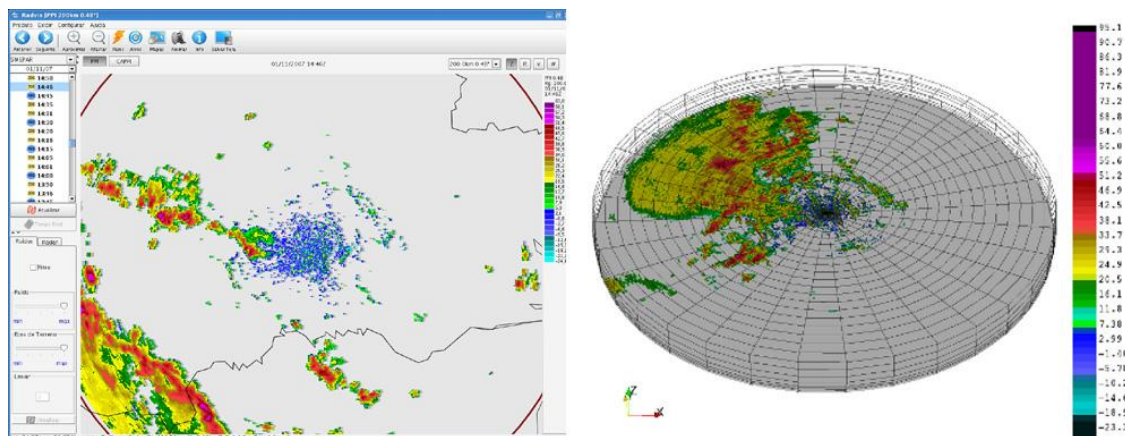


Figura 1 – Uma ferramenta utilizada para visualizar produtos do radar e a estrutura original do volume de dados em uma grade em coordenadas esféricas.

Os produtos mais comumente utilizados para visualização são o chamado PPI (Plan Position Indicator) e o CAPPI (Constant Altitude Plan Position Indicator). Em PPI a variável é plotada num plano onde o centro indica a posição do radar. O plano é fixo para um determinado ângulo de elevação, portanto deve-se observar que conforme aumenta a distância em relação ao radar, a altitude também aumenta. Em CAPPI a visualização de uma variável a uma elevação constante, ou seja, em um plano paralelo à superfície terrestre. Diferentemente do PPI, em uma imagem de CAPPI, a qualquer distância do radar os pontos estarão na mesma altura. A Figura 2 mostra o esquema par obtenção dos produtos CAPPI.

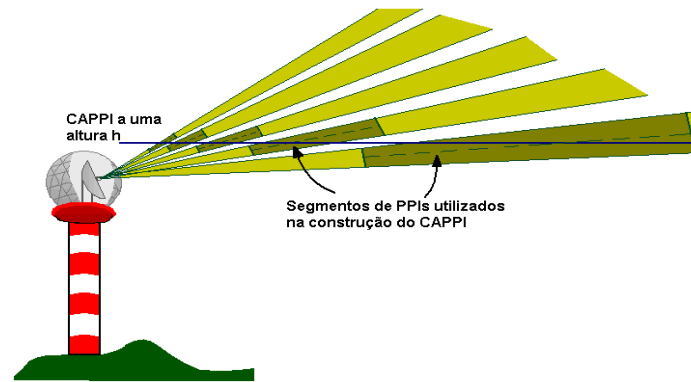


Figura 2 - Esquema para obtenção dos produtos CAPPI e PPI.

3. DESENVOLVIMENTO DE UM VISUALIZADOR INTERATIVO EM PYTHON

Utilizou-se a linguagem Python e os pacotes Numpy e Pygame para desenvolver um visualizador de dados de radar meteorológico, focado na facilidade de uso e na velocidade de resposta do sistema às solicitações do usuário. O objetivo foi proporcionar um meio efetivo de exploração e análise visual dos dados por meio de funcionalidades e ferramentas possam oferecer ao usuário uma experiência mais agradável e efetiva na análise espacial dos dados.

No sistema desenvolvido, a interação ocorre de maneira gráfica e um PPI serve como ponto de partida para a definição dos parâmetros usados nos algoritmos de visualização. Por exemplo, a partir da tela inicial, se o usuário manter pressionado o botão esquerdo do mouse e mover o cursor sobre a representação dos dados, uma linha para o corte vertical será mostrada. No instante em que o usuário solta o botão o algoritmo que processa os dados para gerar o corte é executado e o corte é apresentado em uma fração de segundo.

A abordagem utilizada para conferir uma resposta rápida na geração do corte vertical foi calcular a posição espacial de cada pixel da imagem que representa o corte e, então, buscar nos dados o valor da variável correspondente àquela posição. Essa estratégia é conveniente para a utilização do pacote Numpy, pois as operações com matrizes podem ser escritas com poucas linhas de programação e desempenho satisfatório. Não é objetivo deste trabalho apresentar testes de velocidade de processamento, no entanto, é possível estimar que em um PC comum, um tempo menor que 1/5 segundos, o que é aceitável para nosso propósito.

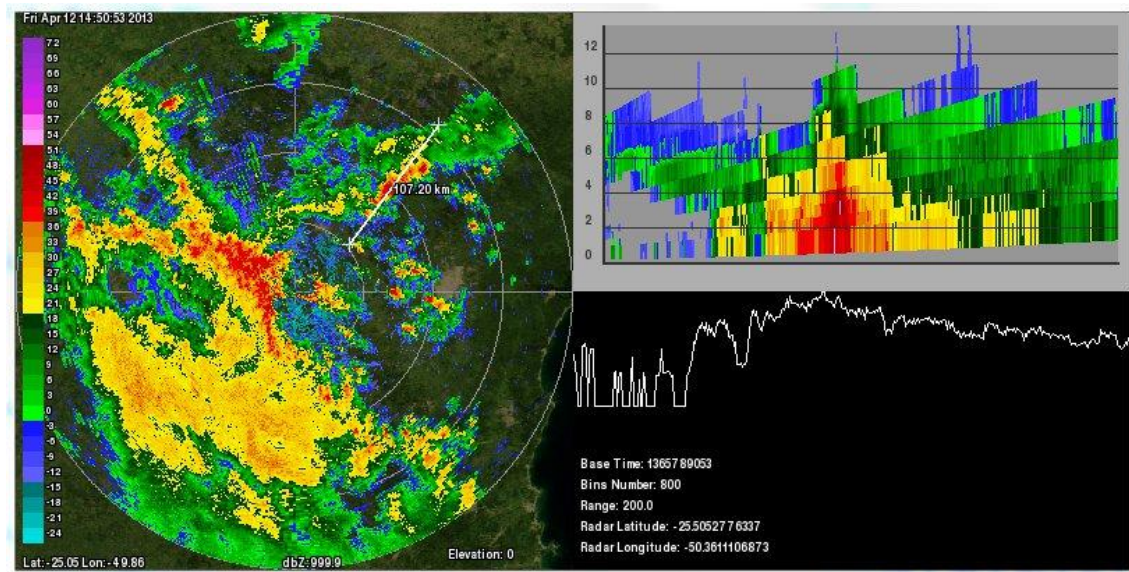


Figura 3 – Tela do visualizador de dados desenvolvido

Outra característica do aplicativo desenvolvido é que a representação dos dados pode ser ampliada (zoom in) interativamente, em tempo real, utilizando a rodinha do mouse e mantendo o cursor no centro área de interesse. A ampliação da imagem é apresentada imediatamente utilizando as funções do Pygame para aplicar um fator de escala na imagem. Após a interação cessar, a representação dos dados é atualizada, a partir dos dados originais, mostrando-os em sua estrutura polar original. É comum, em aplicativos para visualizar este tipo de dado, a conversão da estrutura polar em uma grade cartesiana, então, ao ampliar uma área a resolução e configuração original é perdida. A Figura 4 mostra a ampliação de uma região em que manteve-se a estrutura original dos dados (figura da esquerda) em comparação a uma ampliação realizada sobre uma região previamente convertida para o sistema cartesiano (figura da direita).

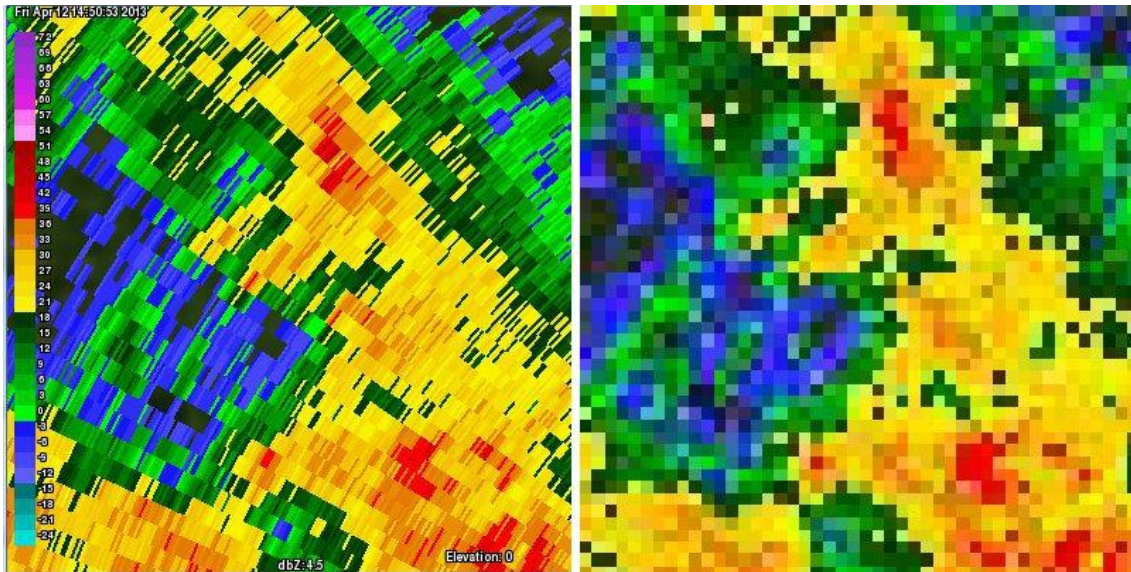


Figura 4 – Visualização ampliada dos dados mantendo a estrutura polar (esquerda) e visualização ampliada dos dados previamente convertidos em estrutura cartesiana.

As estruturas de matrizes e as operações algébricas disponível em Numpy permitem em um única linha de código, por exemplo, realizar operações sobre todos os elementos de uma matriz de forma otimizada. Em um caso específico em que se deseja criar uma grade cartesiana, de resolução 500x500, e converter as coordenadas (x, y) dos pontos em coordenadas polares (r, t), pode-se fazer com as 3 linhas mostrados abaixo.

```
X , Y = numpy.meshgrid(numpy.arange(N) , numpy.arange(N) )
R = numpy.sqrt(X**2 + Y**2)
T = numpy.arctan2(Y,X)
```

4. CONSIDERAÇÕES FINAIS

Como resultado prático deste trabalho foi obtido um aplicativo para visualização de dados de radares meteorológicos que provê funcionalidades personalizadas, não disponíveis em outros programas, comerciais ou não. Os parâmetros para algoritmos de visualização implementados são definidos por meio de interação direta com uma representação dos dados. As estratégias adotadas para a geração das representações dos dados, implementadas fazendo uso intensivo de estruturas de dados baseados em vetores e matrizes, permitiu obter um desempenho satisfatório.

A experiência do desenvolvimento da ferramenta de visualização apresentada neste trabalho, mostrou que a linguagem Python pode ser uma boa escolha para desenvolvimento de sistemas gráficos que combinam processamento numérico e interatividade. Embora em computação científica linguagens como C/C++ e Fortran sejam mais utilizadas, linguagens como Python oferecem muitas vantagens em termos de produtividade e podem alcançar desempenho satisfatório para muitas aplicações. As facilidades e conveniências providas pelas bibliotecas e módulos disponíveis em Python, permitem a criação de programas de forma mais rápida e fácil.

REFERÊNCIAS

- [1] F. C. Coelho, *Computação Científica com Python*. 1th. Ed. [s.l.]: flávio codeco coelho, 2007. 305 p.
- [2] R. Jones. *Rapid Game Development In Python*. OpenSource Developers' Conference (2005), 84-90.
- [3] M. Tory, T. Moller. "Human factors in visualization research," *Visualization and Computer Graphics*, IEEE Transactions on , vol.10, no.1, pp.72,84, Jan.-Feb. 2004
- [4] L.J. Wei-Bing, 2012: *Why Python Is the Next Wave in Earth Sciences Computing*. Bull. Amer. Meteor. Soc., 93, 1823–1824.
- [5] Q. Wen. *3D Interaction with Scientific Data: An experimental and perceptual approach*. PhD thesis. Technische Universiteit Eindhoven, 2008.