

Panoramas perfeitos com Hugin, Enblend e Autopano-SIFT

Visão de 360 graus

É possível criar uma imagem agradável alinhando múltiplas paisagens para formar um panorama. Mas o alinhamento pode ser cheio de manhas, especialmente se as fotos não combinarem. Ferramentas como o Hugin, o Enblend e o Autopano-SIFT ajudam a criar um perfeito panorama sem costuras.

POR OLIVER FROMMEL

Pode ser difícil combinar fotos digitais em uma imagem sem costura usando o Gimp [1]. Este artigo mostra algumas ferramentas que ajustam e alinham as imagens automaticamente.

Imagens distorcidas

É fácil tirar uma série de fotos digitais para construir uma imagem panorâmica, mas é aí que o problema começa. É muito difícil obter emendas suaves e invisíveis. Normalmente, as fotos estão tão distorcidas que é impossível corrigi-las manualmente. Parte da culpa é da lente; ela geralmente distorce as imagens, transformando linhas retas em curvas que dificultam a composição da foto.

A outra razão é a distorção causada por se tirar uma série de fotografias. Se você ficar parado e virar a câmera para a direita e a esquerda, os objetos à direita da imagem parecerão muito grandes, enquanto os que estão à esquerda aparecerão muito pequenos.

Há software livre que pode ajudar a corrigir esse tipo de distorção, alinhar as imagens e, finalmente, montá-las para criar um panorama.

Equalizando e alinhando fotos

O programa gráfico Hugin [2] exibe as imagens, permitindo ao usuário selecionar os pontos problemáticos para equalizá-las e alinhá-las. O programa é fácil de instalar, mas requer algumas bibliotecas não incluídas em certas distribuições. Veja mais detalhes no **quadro 2: Instalação**.

Depois de digitar `hugin` para iniciar o programa, surge a janela principal mostrada na **figura 1**. Clique em `Add individual images...` para selecionar cada imagem. Na caixa de diálogo que aparece, você pode segurar o **Ctrl** e clicar em diversas imagens para selecioná-las.

O Hugin exibe uma lista de imagens no painel da esquerda, acrescidas de alguns detalhes, como largura e altura; valores rotacionais (**Yaw**, **Pitch**, **Roll**) também são exibidos, embora o padrão seja zero. O Hugin usa a primeira foto como referência para alinhar as outras imagens. O botão `Anchor this image...` permite selecionar outra imagem como ponto de referência.

A segunda aba, **Camera and Lens** (**figura 2**), serve para corrigir a distorção ótica causada pela câmera e suas lentes. Se suas imagens tiverem informações EXIF (*Exchangeable Image File Format* – Formato

Quadro 1: PanoTools

Por muito tempo, o pacote PanoTools foi o único software livre para criar imagens panorâmicas. As condições de licenciamento do PanoTools sempre foram complexas. A biblioteca e o *PTOptimizer* foram lançados sob a licença livre *GNU Public License*, mas diversas outras licenças se aplicam a alguns dos programas individuais.

Como o autor do programa, Helmut Dersch, não tem mais o software em sua própria homepage, outras pessoas estão oferecendo pacotes antigos. Na verdade há até mesmo alguns derivados comerciais, embora talvez não sejam exatamente o que se pode chamar de legais.

Desde então, um projeto foi fundado no SourceForge para manter as partes livres do PanoTools [3].

de arquivo de imagem intercambiável), o Hugin pode extrair automaticamente os dados necessários. Praticamente todas as câmeras digitais atuais adicionam essas informações às imagens. Se você não sabe qual o ângulo de visão de suas lentes, consulte a **tabela 1** (tirada da documentação do PanoTools).

A opção *Inherit* permite que todas as outras imagens herdem as configurações da imagem atual. Se essa configuração for desabilitada, o programa pode lidar com imagens que tenham diferentes valores de distorção, útil quando as imagens foram capturadas com ajustes ou lentes diferentes. Após descobrir os melhores valores para sua câmera (provavelmente por tentativa e erro), você pode armazená-los em *Lens Management* e usá-los em seus futuros projetos.

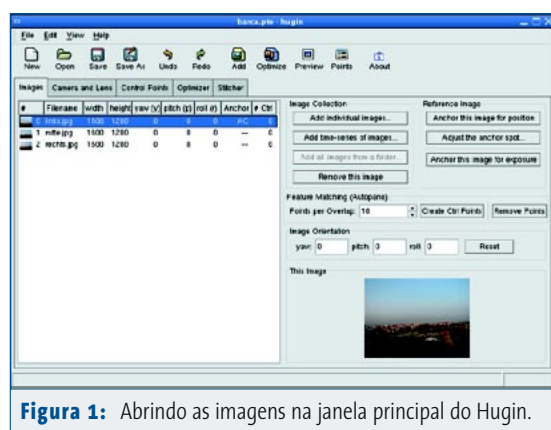


Figura 1: Abrindo as imagens na janela principal do Hugin.

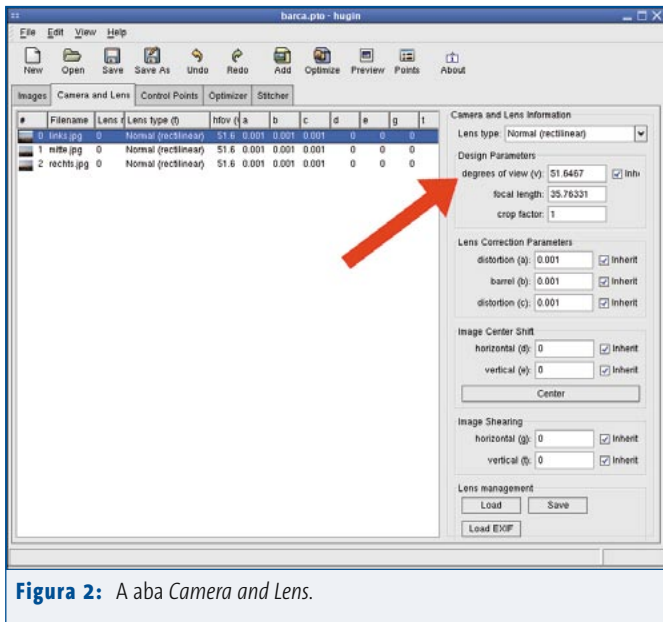


Figura 2: A aba Camera and Lens.

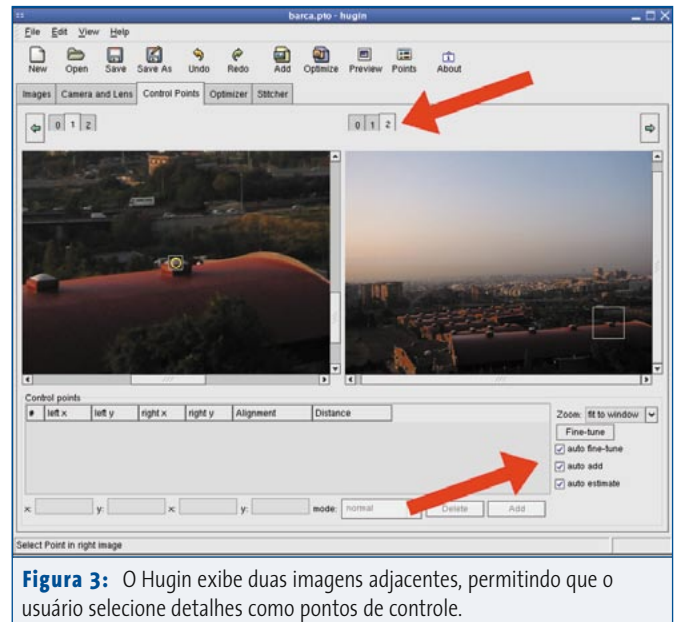


Figura 3: O Hugin exibe duas imagens adjacentes, permitindo que o usuário selecione detalhes como pontos de controle.

Pontos de Controle Manuais

O Hugin precisa de alguns detalhes definitivos da imagem para conseguir alinhar corretamente duas imagens adjacentes. A aba *Control Points* cuida disso. Essa aba mostra duas imagens e pede que você clique em pontos significativos para criar atribuições.

Primeiro, faça com que o Hugin exiba duas imagens adjacentes ajustando o número da aba acima da imagem da direita para um número maior que o da aba esquerda (ver **figura 3**). Em outras palavras, se ambos forem 0 quando você começar, a aba direita precisa ser configurada como 1. Em seguida use as setas direcionais para passar de uma imagem à outra e o Hugin mostrará as imagens consecutivas à direita e à esquerda.

Você pode querer selecionar *auto fine-tune* e *auto-estimate*, pois essas opções fazem com que o Hugin ajude você a escolher os pontos de controle. O *auto add* diz ao programa para adicionar cada par mapeado nos pontos específicos – dá para fazer isso manualmente com o botão *Add*.

Se você tiver pontos suficientes, pode passar para a etapa de “ligar os pontos” clicando em *Optimizer* (**figura 4**). Marque a opção *Positions* (*pairwise optim.*, *star-*

ting from Anchor) para o *Optimizer* e em seguida clique em *Optimize now!*. As três caixas de texto abaixo exibem o número da imagem e o valor pelo qual o Hugin precisa corrigi-la, segundo os colchetes assinalados. Se uma imagem não estiver assinalada, isso significa que o Hugin não vai alterá-la, o que será o caso da imagem de referência (*Anchor*). Se você selecionar a opção *the Custom parameters*

below para o *Optimizer*, pode configurar manualmente os valores para cada imagem – essa configuração é indicada para os especialistas no assunto.

O botão *Preview* na barra de ferramentas traz uma previsão com os valores calculados. Você pode selecionar *Auto* na janela de visualização para fazer com que o programa mostre automaticamente os resultados mais recentes. ➔

Quadro 2: Instalação

O Hugin precisa da biblioteca *wxWidgets* (anteriormente chamada *wxWindows*) e da extensão *wxWidgets-XRC*. A maioria das distribuições inclui as outras bibliotecas necessárias, como *Libtiff*, *LibPNG* e *LibJPG*. Além do pacote matemático *FFTW*, o Hugin também precisa da biblioteca *Boost*, se a intenção for compilá-lo você mesmo. Infelizmente, a versão exata e o nome do pacote dependem da distribuição. Por exemplo, a biblioteca de desenvolvimento *Boost* para o Fedora é chamada *libboost-devel*; já os usuários do Debian precisam de três pacotes chamados *libboost-graph-dev*, *libboost-dev* e *libboost-python-dev*.

No Fedora, o seguinte comando instala as bibliotecas necessárias:

```
rpm -iv panorama-tools-2.7a-1.i386.rpm
wxGTK-* fftw2-*
```

Entre as ferramentas de panorama está o *libpano12*, mas isso não impede que o gerenciador de pacotes se queixe de um pacote faltante – já que as dependências dos pacotes estão incorretas. Para contornar isso é preciso especificar uma opção de programa RPM para desabilitar a checagem de dependências: `rpm -iv --nodeps hugin-0.4-cvs20041021.bp.fc2.i386.rpm`.

Embora o SuSE traga por padrão uma versão do *wxWindows*, o Hugin não pode usá-lo. Assim, será preciso remover o pacote existente e usar os do Fedora – não se desespere, isso funcionou muito bem com o SuSE 9.1 em nosso laboratório.

```
rpm -iv panorama-tools-2.7a-1.i386.rpm
wxGTK-xrc-2.4.2-4.1.fc2.dag.i386.rpm
fftw-2.1.3-1102.i586.rpm hugin-20040921-
1.bp.fc1.i386.rpm wxGTK-2.4.2-4.1.fc2.
dag.i386.rpm
```

O Debian não tem pacotes *wxGTK* adicionais que incluam *XRC* nos nomes dos pacotes; em vez disso, procure por *libwxgtk2.4-contrib-dev*. Você pode usar os seguintes comandos para instalar as bibliotecas necessárias, o *LibPano* e o *Hugin* :

```
apt-get install libwxgtk2.4-dev
libwxgtk2.4-contrib-dev fftw2
dpkg -i libpano12_20040917-1_i386.deb
dpkg -i hugin_0.4pre20040917-1_i386.deb
```

Isso poderá funcionar ou não, dependendo do programa e das bibliotecas instalados anteriormente. Se ainda estiver faltando algum pacote, tente procurar por ele no Google, usando como palavras-chave o nome do pacote e o formato necessário. Veja mais detalhes sobre os requisitos nos quadros do *Autopano* e do *Enblend*.

Tabela 1: Lentes e Campo de Visão

Comprimento Focal	Campo de Visão Paisagem	Campo de Visão retrato
28mm	65	46
35mm	54	38
50mm	40	27

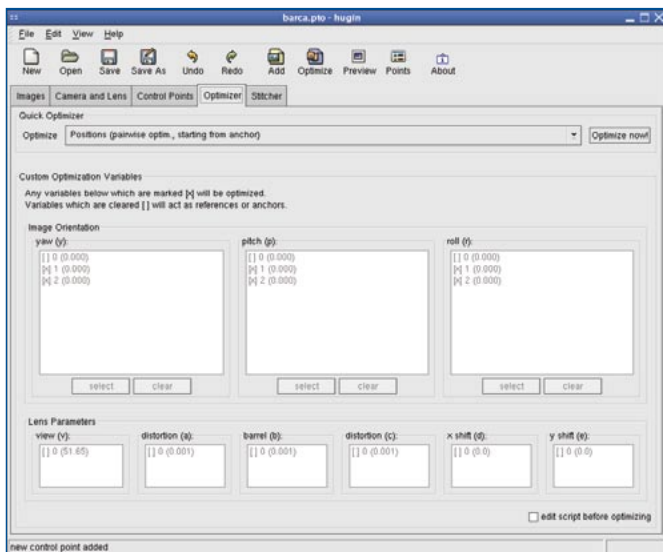


Figura 4: A aba *Optimizer* permite o controle granular dos valores de otimização estabelecidos automaticamente pelo Hugin.

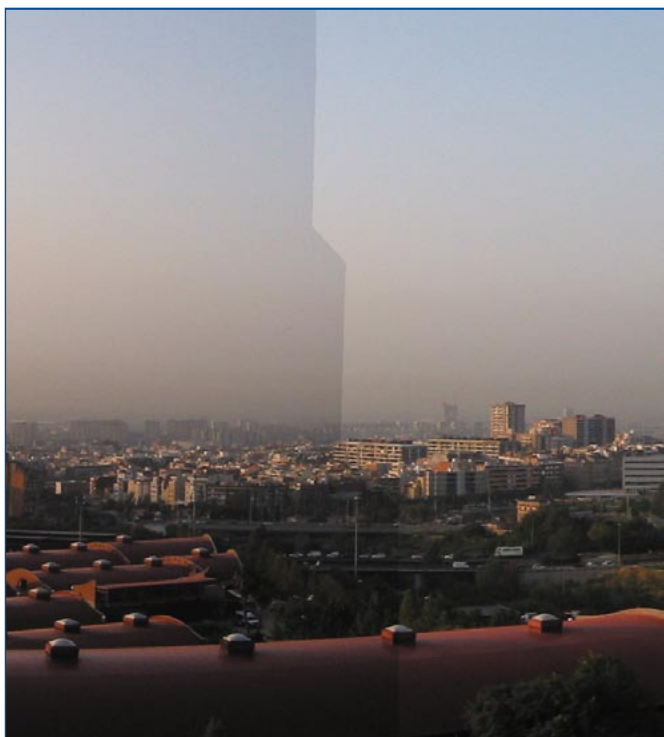


Figura 5: As imagens individuais estão na ordem correta, mas as emendas estão claramente visíveis.

GLOSSÁRIO

bytecode é um formato intermediário de compilação, no qual o programa não é compilado para um hardware específico mas para uma **máquina virtual**. Os programas em bytecode podem ser considerados multiplataforma, pois rodam praticamente em qualquer hardware que tenha uma máquina virtual daquela linguagem. O exemplo mais comum de utilização de bytecodes, além do Mono, é na linguagem Java, em que o mesmo programa pode rodar – sem recompilação – no Windows®, no Unix/Linux e no MacOS, desde que a máquina virtual Java esteja instalada.

Roll, Pitch and Yaw são termos para rotação espacial em torno de três eixos. *Roll* descreve uma rotação sobre um eixo perpendicular ao plano da imagem, *pitch* é a rotação sobre um eixo horizontal e *yaw* é a rotação sobre um eixo vertical.

Montando Imagens

A última aba, o *Stitcher*, é o coração do programa – e o maior desafio ao mesmo tempo, pois as coisas nem sempre funcionam como se espera. *Stitching* (costura) é o processo de reunir muitas imagens para criar uma imagem panorâmica. Esse processo tipicamente implica o alinhamento das imagens e a correção antecipada das distorções. O Hugin pode trabalhar com dois mecanismos de *stitching*, os módulos do software que realmente montam a imagem panorâmica. O *Nona* é um “costureiro” fornecido com o próprio Hugin. Funciona bem, embora a qualidade não possa ser considerada muito melhor do que um “rascunho”. Na maioria dos casos, as emendas podem ser vistas no produto final (como no caso da **figura 5**).

Antes de passar as imagens na máquina de costura, certifique-se de haver selecionado *Calculate Field of View* para recalcular o campo de visão. Apenas os autores do programa sabem por que o Hugin não faz isso automaticamente. Da mesma maneira, será preciso modificar o tamanho da imagem selecionando *Calculate Optimal Size*. Se você estiver usando o mecanismo *Nona* que vem com o programa, o formato *JPG* já estará pré-selecionado e bastará clicar em *Stitch now!*. Pouco tempo depois (ou possivelmente muito tempo depois, dependendo do tamanho e do número de imagens), você deverá ter a imagem final em seu disco rígido com o nome escolhido.

A alternativa é usar o programa externo *Enblend* [4], que o Hugin não suporta diretamente e que exige uma etapa a mais. Primeiro, use o Hugin para criar uma série de imagens TIFF, que estejam corretamente alinhadas mas não alinhadas. Escolha então a opção *into a layered TIFF file* para *Stitch the images*. O programa então selecionará automaticamente *Multiple TIFF* como formato de saída. Clique em *Stitch Now!* e o Hugin pedirá um nome de arquivo, acrescentando um número de quatro dígitos ao nome de cada imagem. Você pode então especificar o nome de saída do arquivo:

```
enblend -o panorama.tif pic0000.tif pic0001.tif pic0002.tif
```

Os resultados são bastante impressionantes. O *Enblend* faz o *stitch* das imagens sem deixar visíveis aquelas costuras feias.

Automatização dos pontos de controle

Se isso tudo exige “clicação” demais para o seu gosto, você pode usar outro programa para configurar automaticamente os pontos de controle. O *Autopano-SIFT* [5] pode até mesmo encontrar combinações invisíveis para os olhos humanos em suas imagens. Para rodar a ferramenta, chame o script e não se esqueça de especificar o arquivo de saída (usando o parâmetro *-o*), bem como as imagens separadas. O Hugin pode, posteriormente, interpretar o arquivo de saída.

```
autopano-complete.sh -o saida esquerda.jpg centro.jpg direita.jpg
```

Como alternativa, você pode preferir usar o front-end gráfico *autopanog.exe*, que faz exatamente a mesma coisa, mas ao menos o poupa do trabalho de digitar os nomes de arquivo. Ao abrir o arquivo PTO resultante com o Hugin, é preciso ajustar os *degrees of view* na aba *Camera and Lens*. As etapas do *Optimizer* e do *Stitcher* já foram descritas anteriormente.

Quadro 3: Enblend

O pacote *Enblend* [6] não precisa de nenhuma biblioteca especial, exceto a *libtiff* e a *libjpeg*. No Fedora e no SuSE, pode-se instalar o *Enblend* digitando `rpm -iv enblend-2.0-1.bp.fc2.i386.rpm`.

Infelizmente, os binários não estão disponíveis para todas as distribuições e pode ser necessário compilar você mesmo o *Enblend*, o que exige a instalação de mais software. A linha a seguir cuida da instalação no Debian:

```
apt-get install libjpeg62-dev libtiff4-dev libpng3-dev fftw-
dev libboost-graph-dev libboost-dev libboost-python-dev
```

Passa então para a compilação do *Enblend*.

Para onde agora?

Essas dicas devem ajudá-lo a começar. O fato de não termos nem mesmo citado grande parte da enorme gama de opções e menus serve bem para mostrar o potencial do programa. Podemos afirmar com segurança que os problemas menores serão resolvidos muito em breve. E se você tiver vontade de ajudar nessa tarefa após ler este artigo, por que não? ■

INFORMAÇÕES

- [1] Gimp, the GNU Image Manipulation Program: <http://www.gimp.org>
- [2] Hugin: <http://hugin.sourceforge.net/>
- [3] Nova homepage para os órfãos do PanoTools: <http://panotools.sourceforge.net/>
- [4] Enblend: <http://enblend.sourceforge.net/>
- [5] Autopano-SIFT: <http://user.cs.tu-berlin.de/~nowozin/autopano-sift/>
- [6] Mono: <http://www.mono-project.com/downloads/>
- [7] Mono no Debian: <http://pkg-mono.alioth.debian.org/>
- [8] Pacotes para diversas ferramentas de panorama: <http://bugbear.blackfish.org.uk/~bruno/panorama-tools/>
- [9] GPhoto, software para download de imagens de câmeras digitais: <http://www.gphoto.org>
- [10] DigiKam, interface KDE para o GPhoto: <http://digikam.sourceforge.net>
- [11] F-Spot, aplicativo para download e gerenciamento de imagens para o Gnome: <http://www.gnome.org/projects/f-spot/>

Quadro 4: Mono para o Autopano

O *Autopano-SIFT* [5] foi escrito na linguagem de programação C# e portanto precisa das bibliotecas do projeto Mono [6]. Para usar o programa, não é necessário o ambiente Mono completo. Você precisará do interpretador, que roda o programa, e algumas bibliotecas. Os pacotes no SuSE são `mono-core-1.0.2-1.ximian.9.1.i586.rpm`, `libicu26-2.6.2-1.ximian.9.0.i586.rpm`, `gtk-sharp-1.0.2-1.ximian.9.2.i586.rpm`, `libgtkhtml3.1_7-3.1.13.0.200405120525-0.snap.ximian.9.1.i586.rpm`. Os usuários do Fedora não precisam do pacote `libgtkhtml`.

Os usuários do Debian devem acrescentar o repositório APT ao arquivo `/etc/apt/sources.list` e instalar os pacotes `mono`, `libgtk-cil` e `libglib-cil` pela linha de comando. O website do Mono no Debian [4] traz mais detalhes.

Arquivos `exe` escritos em Mono são executados com o comando `mono programa.exe`. Um módulo do kernel chamado *Binfmt-Misc* pode executar programas em **bytecode** do Mono diretamente. A maioria das distribuições inclui o módulo por padrão. Se seu *automounter* não carregar o pseudo-sistema de arquivos exigido para o uso dessa funcionalidade, tente o seguinte:

```
mount -t binfmt_misc none /proc/sys/fs/binfmt_misc
```

Finalmente, você precisa dizer ao kernel como reconhecer arquivos do Mono e rodar `mono`:

```
echo ':CLR:M::MZ::usr/bin/mono:' >
/proc/sys/fs/binfmt_misc/register
```

Supondo que seus arquivos `exe` tenham permissão para execução (`chmod +x`), você pode iniciá-los diretamente. Ainda será preciso descompactar o *Autopano-SIFT* [5]. Se o módulo *Binfmt-Misc* estiver em uso, basta rodar o programa diretamente

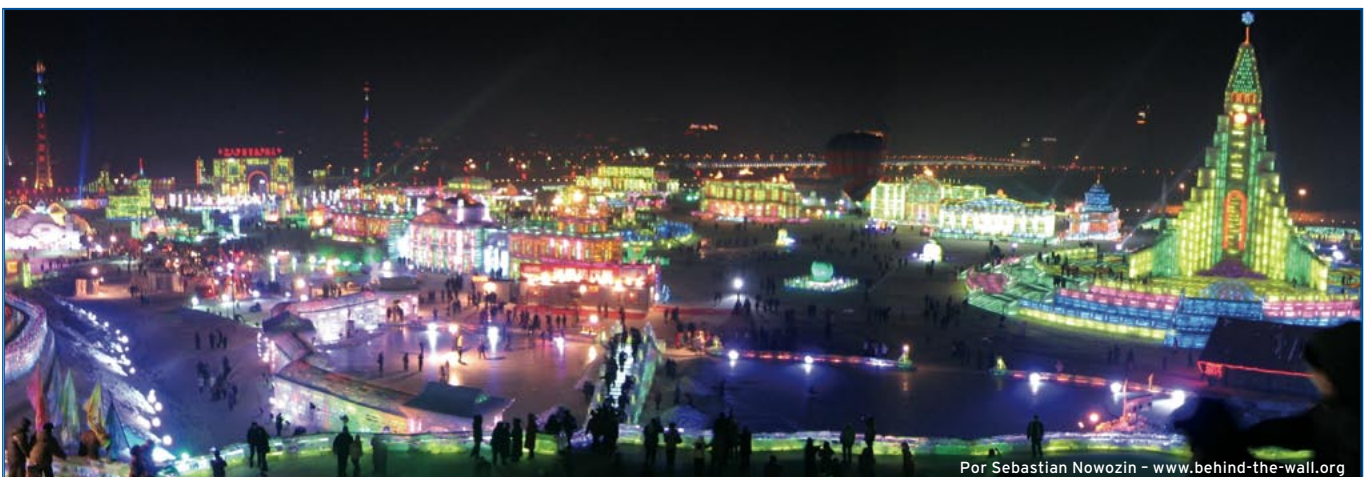
```
/caminho/até/autopano-sift-2.0/bin/autopano.exe
```

Se isso falhar, use o interpretador do Mono:

```
mono /caminho/até/autopano-sift-2.0/bin/autopano.exe
```

Se tudo der certo, o *Autopano* deve exibir um curto texto de ajuda. Se o subdiretório `bin` do *Autopano* estiver em sua variável de ambiente `PATH` (ou seja, for definido como o diretório onde o sistema busca executáveis), não é preciso digitar o caminho completo. Essa é também a melhor maneira de testar a versão gráfica, `autopanog.exe`.

Para permitir que o script `autopano-complete.sh` funcione, é preciso configurar sua variável `PATH` como descrito, já que de outra forma o script reclamará que não conseguiu encontrar um “assembly”.



Por Sebastian Nowozin - www.behind-the-wall.org

Figura 6: Detalhe de um panorama do festival do gelo de Harbin, na China, gerado com o *Autopano-SIFT*.