

Ambiente Integrado para Posicionamento em Operações Militares

GUSTAVO MOREIRA PIERRE¹, MARCELO GATTASS², ROBERTO DE BEAUCLAIR SEIXAS^{2,3}

¹Centro de Instrução Almirante Sylvio de Camargo, Marinha do Brasil, Rua Magno Martins, Rio de Janeiro, RJ, Brasil
gustavopierre@uol.com.br

²Tecgraf–Pontifícia Universidade Católica, Rua Marquês de São Vicente, 225, 22453-900 - Gávea, Rio de Janeiro, RJ, Brasil
gattass@tecgraf.puc-rio.br

³IMPA–Instituto de Matemática Pura e Aplicada, Estrada Dona Castorina, 110, 22460 Rio de Janeiro, RJ, Brasil
tron@impa.br

Abstract. With technological development, we have been witnessing a growing popularization of satellite-location equipment (GPS receivers) and pocket computers (PDA) in several areas: air, sea and earth transports, mapping and geoprocessing, sports and entertainment. In the military environment, this popularization is also occurring: the Armed Forces in several countries are obtaining increasing numbers of GPS receivers and are testing equipment to interact with them. The purpose of this work is to integrate GPS receivers and PDAs, with the intention of employing them in military operations in order to obtain quick responses for a faster and more efficient use of fractions of involved troops. This work involves the generation of files readable by PDAs, the communication of GPS receivers with PDAs, and an easy and fast visualization of the information provided by the receivers. It also includes a comparison between the use of maps in two different representations, raster and vector, both allowing for zooming and movement of the image, as well as the possibility of measuring distances between points of the map in the PDA screen, and the inclusion, exclusion and alteration of several observations.

1 Introdução

O emprego de equipamentos portáteis nas mais diversas áreas tem presenciado um crescimento vertiginoso, principalmente em função do grande crescimento tecnológico. Nos últimos dez anos, vimos uma popularização crescente deste tipo de equipamento também no meio militar, onde os quartéis tendem a adquirir cada vez mais equipamentos de posicionamento por satélite (receptores GPS) e testam equipamentos para interação com os mesmos.

A tecnologia dos portáteis está em franca ascensão, o que faz surgir constantemente novos softwares. Muitos destes não têm uso militar. Os que possuem características militares normalmente acompanham algum equipamento que é comercializado, normalmente, a preço elevado.

O objetivo principal deste trabalho [3] é um modelo de ambiente integrado, envolvendo hardware e software, para posicionamento em operações militares. As metas deste trabalho são: geração, no computador pessoal (PC), de um arquivo contendo o mapa *raster* ou vetorial e outras informações, que possa ser manuseado pelo PDA; análise do formato do mapa mais adequado para manuseio pelo PDA, se *raster* ou vetorial, em relação à transferência, armazenamento e operação, levando em consideração as limitações de armazenamento e performance do PDA; acompanhamento, no mapa, da posição fornecida pelo receptor GPS; e possibilidade de interação do usuário com o PDA para, por exemplo, fazer observações e

medir distâncias.

O trabalho de Steve Mann e Ray Rischpater [2] não utilizam mapas de tamanho superior aos 400x400 pixels, o que não permite a utilização de um mapa envolvendo toda uma área de operações. Por exemplo, um mapa na escala de 1/50.000, abrangendo uma área de operações de dimensões de 20 x 20 quilômetros, apenas cerca de 10% dele pode ser visto, com o devido detalhamento, numa área de 400 x 400 pixels. Isso não diminuiu a excelência do trabalho destes no que se refere a uma série de conceitos apresentados e técnicas de zoom e movimentação de mapas no PDA.

2 Utilização de Dispositivos Portáteis para Posicionamento em Operações Militares

2.1 Soluções

Tendo em vista que o objetivo do trabalho é um modelo de ambiente integrado, envolvendo hardware e software, para posicionamento em operações militares, essa integração foi atingida com uso de soluções que serão agora descritas.

2.1.1 Hardware

O hardware envolvido neste trabalho é constituído de: computador pessoal para a geração, a partir de um mapa *raster* ou vetorial, de um arquivo de dados para ser transferido e, posteriormente, lido pelo PDA; receptor GPS GARMIN eTrex, que fornece as informações de hora e posição

para o PDA; PDA Workpad C505 da IBM (versão OEM do Palm m505) para visualizar o mapa, oriundo do computador pessoal, e obter informações do receptor GPS; e cabos e conectores responsáveis pela ligação entre os equipamentos.

Como o software no PC visa somente transformar os mapas nos formatos legíveis (PRC e PDB) ao Sistema Operacional da Palm (PalmOS), não exige muito poder de processamento nem muita memória do PC.

A opção de escolha do receptor GPS GARMIN eTrex se deu devido principalmente a possibilidade de emissão dos dados utilizando o protocolo NMEA0183 e de envio desses dados para um equipamento conectado a ele por meio de uma porta serial RS232, além de permitir funcionar em modo simulado. Estas características possibilitaram o desenvolvimento de um ambiente que pode fazer uso de outras marcas de receptores GPS com características semelhantes, bem como a realização de testes com receptor GPS conectado ao PC sem a necessidade de transportar este para uma área que permitisse a recepção de informações dos satélites.

O custo do Workpad C505, que utiliza o sistema operacional da Palm, a facilidade de explicações, tutoriais e exemplos gratuitos principalmente na internet, além da possibilidade de conexão com a porta serial RS232 de um equipamento para recebimento e envio de dados foram fatores decisivos na escolha deste PDA. O problema deste equipamento é sua vulnerabilidade à quedas e à água, mas essa resistência só se consegue com equipamentos cujo preço atinge valores de oito a dez vezes o preço de um equipamento comum.

2.1.2 Software

Como parte deste trabalho foi implementado um programa que gerasse o arquivo de banco de dados que é transferido para o Palm e nele manipulado.

Geração do Banco de Dados

Esta aplicação tem duas funções básicas: ler um arquivo de projeto (de extensão PRJ) contendo dados de vários arquivos de mapas vetoriais (formato *shape files*) e gerar um ou mais arquivos PDB contendo os dados para reconstrução destes mapas; e ler um arquivo de projeto (de extensão PRB) contendo dados de vários arquivos de mapas *raster* (formato windows bitmap) e gerar um único arquivo PRC contendo os recursos tipo bitmap correspondente aos mapas.

Tanto a geração do arquivo PDB como do arquivo PRC foram feitos byte a byte. A ordenação de gravação foi importante, pois eles teriam que ser gravado no esquema denominado *big-endian* [1].

A partir de agora será especificada a criação do

arquivo PDB e a criação do arquivo PRC.

Geração do PDB

O programa implementado converte os arquivos *shape files* (vetoriais) em arquivos PDB. Algumas simplificações foram feitas para a geração do PDB: não foram considerados os atributos constantes no arquivo dBASE, nem o arquivo de índices; não foram considerados os buracos nos polígonos vazados; os polígonos foram considerados como seqüências de linhas sem preenchimento, uma vez que o sistema operacional da Palm nativamente possui apenas quatro elementos gráficos de desenho (ponto, linha, retângulo - que pode evoluir para um círculo - e bitmap); e, por fim, o programa converte, no máximo, dois arquivos *shape files* em PDB - esta simplificação visa somente uma facilidade na geração dos resultados para avaliação, ou seja, apesar de normalmente serem necessárias mais camadas de informação vetorial, apenas duas foram utilizadas visando a medição do tempo de sua manipulação para comparação com mapas *raster*.

Como um mesmo arquivo *shape file* deve conter feições do mesmo tipo, para a representação de um mapa são necessários normalmente mais de um arquivo *shape file*, um para cada feição. Em virtude disso houve a necessidade de se criar um arquivo de projeto, que nada mais é senão um arquivo ASCII cujo nome possui extensão PRJ e cada linha é um arquivo *shape file*, um espaço e o nome da cor da feição.

O programa lê cada arquivo *shape file* e define uma caixa envolvente única para ambos. A partir daí, diferente do arquivo *shape file*, ele não considerará a caixa envolvente de cada objeto de cada arquivo *shape file*. São gerados então, para cada arquivo *shape file*, que será lido novamente, o cabeçalho, a lista de registros e, finalmente, os registros.

Sendo, de uma forma geral, os arquivos PDB constituídos de uma série de registros contíguos, foi adotado que o primeiro registro do arquivo PDB gerado deve conter a quantidade de objetos, o tipo dos objetos e a caixa envolvente única, enquanto cada um dos seguintes registros deve representar um objeto. Todos os campos são convertidos para caracteres antes de serem gravados e são separados por um espaço.

Geração do PRC

Antes de começar a explicação do processo de geração do arquivo PRC, é importante que se saiba que o tipo de arquivo bitmap utilizado foi o arquivo no formato window bitmap sem compactação e com oito bits de profundidade de cor. A compactação RLE, apesar de ser adotada de maneira simples pelo PalmOS, não foi utilizada devido à incompatibilidade com o mesmo processo realizado em softwares gráficos populares para PC. A outra simplificação é a

utilização de bitmaps com oito bits de profundidade de cor, que é suficiente para representar um mapa para operações militares.

O uso de recursos bitmap no lugar de registros contendo as informações de cada pixel da imagem é porque o desenho pixel a pixel é bem mais lento do que o desenho do bitmap como um recurso.

Tendo em vista que as imagens utilizadas para orientação durante as operações militares requerem um certo nível de detalhamento que não permitiria colocá-las numa dimensão de cerca de 420 x 420 pixels, conforme limite de dimensões de criação de uma janela em *offscreen*, isto é, em memória (sem exibi-la) ou de um recurso bitmap no PalmOS, houve a necessidade de dividir a imagem bitmap em blocos, onde cada um destes daria origem a um recurso do tipo bitmap. A Figura 1 mostra um exemplo de divisão de uma imagem em blocos de 420 x 420 pixels. O tamanho máximo do bloco tem que se limitar ao tamanho máximo da janela em *offscreen*.



Figura 1: Divisão de uma imagem em blocos.

Devido à possibilidade de deslocamento do mapa (*panning*) e a fim de permitir uma transição suave de um bloco para o outro, deveria haver uma repetição de uma faixa da imagem em cada bloco na dimensão da tela do Palm em relação a cada bloco vizinho, ou seja, cada bloco deveria ter pelo menos faixas de largura ou altura 160 pixels em comum com os blocos que lhe fazem fronteira. Conforme cálculos realizados a utilização de blocos de 420 x 420 pixels com faixas de imagens em comum aumentaria o tamanho do arquivo em até cerca de 140%. A Figura 2 mostra um exemplo de divisão de uma imagem em blocos de 420 x 420 pixels com faixas de imagens em comum com os blocos vizinhos de 160 pixel.

Outra divisão da imagem, e que foi adotada, é em blocos do tamanho da tela, que não geraria repetição de imagens em arquivo. Desta forma cada janela de fundo seria composta de quatro blocos. A Figura 3 representa a divisão de uma imagem em blocos de 160 x 160 pixels e uma área de janela em *offscreen* selecionada. Esse método, conforme os cálculos aumentaria o tamanho do arquivo apenas

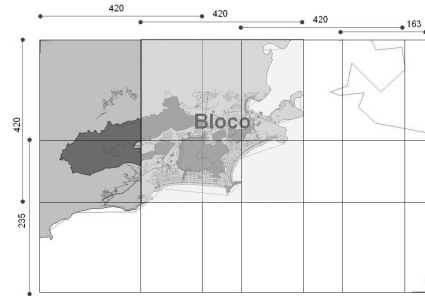


Figura 2: Divisão de uma imagem em blocos com faixas de imagens em comum com os blocos vizinhos.

as em cerca de 6% do seu tamanho original, em virtude do maior número de cabeçalhos (cada recurso bitmap tem um) e da repetição da tabela de cores para cada recurso. O gráfico da Figura 4 representa estes cálculos realizados e permite observar facilmente a diferença entre o tamanho do arquivo gerado utilizando a divisão em blocos de 420 x 420 pixels com faixas de imagem em comum e utilizando a divisão em blocos de 160 x 160 sem faixas de imagens em comum com seus vizinhos em relação ao arquivo original no formato bitmap.

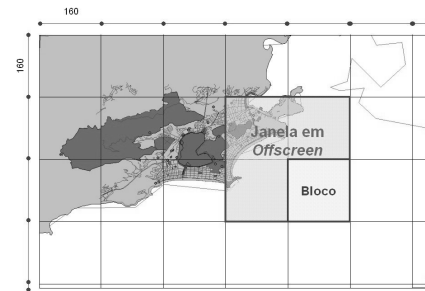


Figura 3: Divisão de uma imagem em blocos de 160 x 160 pixels.

No decorrer do trabalho foi feita uma modificação para aceitar outras imagens no mesmo PRC para indicar os níveis de zoom, uma vez que o zoom conseguido através das APIs do PalmOS deixam a imagem truncada, facilmente verificado nas Figuras 5(a), 5(b), e 5(c).

Para a geração do arquivo PRC foi implementado um programa que lê um arquivo ASCII, com extensão PRB, contendo na primeira linha a quantidade de arquivos bitmap a serem convertidos e nas demais os dados desses arquivos (um em cada linha) separados sempre por um espaço, na seguinte ordem: índice, iniciando em zero; localização (*path*) e nome do arquivo bitmap; e limites mínimo e máximo de longitude e de latitude em UTM, fundamentais para o geo-

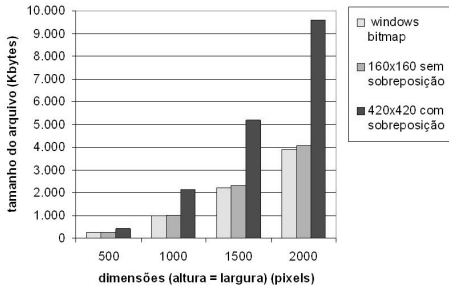


Figura 4: Tamanho do arquivo PRC gerado em relação ao modo de divisão da imagem.

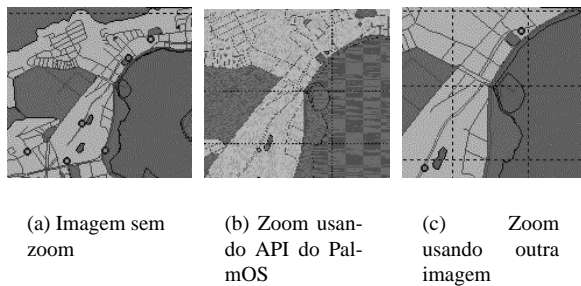


Figura 5: Comparação entre utilização da API do PalmOS e de outra imagem para a operação de zoom

referenciamento da imagem.

Todo o arquivo de projeto PRB dará origem a um único arquivo PRC (com mesmo nome, mas com extensão PRC), ou seja, para cada bitmap será feita uma divisão da imagem em blocos de 160 x 160 pixels e gravados como recursos num único arquivo PRC. Os arquivos bitmap constantes no arquivo de projeto PRB devem estar contidos na área abrangida pelo bitmap de índice zero, ou seja, do arquivo bitmap de índice um em diante deve ser um *zoom in* ou algum detalhamento maior de uma área dentro do bitmap de índice zero. Isso porque quando se executa o *zoom in*, procura-se pela primeira imagem cujos limites estejam dentro dos limites da imagem atual. Se não houver, não será executado o zoom, portanto, a imagem que não estiver dentro dos limites da primeira nunca será usada, estará apenas aumentando o tamanho do arquivo PRC. A Figura 6 ilustra a localização dos bitmaps em relação ao primeiro.

Para guardar informações de número de blocos na horizontal e na vertical de cada bitmap, visando a sua reconstrução, bem como os seus limites mínimos e máximos, visando o georeferenciamento da imagem, foi criado um recurso extra, do tipo texto de informação da aplicação (*appinfostr*), que é basicamente uma cadeia de caracteres contendo o número total de blocos na horizontal, o número total de blocos na vertical, os limites mínimo e

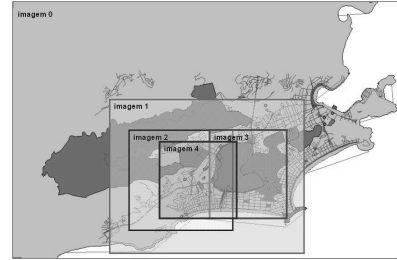


Figura 6: Localização relativa dos arquivos bitmaps de um projeto PRB.

máximo de longitude e de latitude em UTM, todos separados por um espaço. A Figura 7 mostra a distribuição das informações no recurso *appinfostr*.

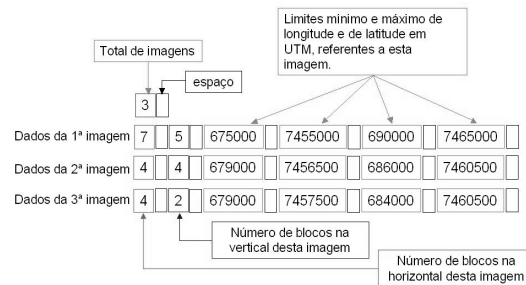


Figura 7: Distribuição das informações no recurso *appinfostr*.

Para a geração do arquivo PRC, será necessário ler os bitmaps do arquivo de projeto PRB três vezes para, em cada uma, realizar as seguintes tarefas: primeiro são definidos e gravados os dados do cabeçalho do arquivo PRC e definido os dados do recurso *appinfostr* (calculado o número de blocos na horizontal e vertical de cada bitmap e lido os limites mínimos e máximos de longitude e latitude de cada bitmap); em seguida é gerada a lista de entradas de recursos; e, finalmente, os bitmaps são divididos em blocos e gravados como recursos e é gravado o recurso *appinfostr*.

No cabeçalho de cada recurso tem o índice, que é fundamental para identificar o bloco na imagem como um todo, sendo assim necessário para a montagem da imagem e também para a obtenção de uma imagem quando for executada uma operação de zoom. O índice tem que ser único para recursos do mesmo tipo. A padronização adotada é a seguinte: na primeira imagem o índice do primeiro bloco do canto superior esquerdo é 5.000, à medida que se anda na horizontal vai acrescentando 1 no índice de cada bloco e a medida que se anda na vertical acrescenta 100 no índice de cada bloco; o primeiro bloco

da imagem seguinte começa em 10.000, a seguinte 15.000 e assim por diante, ou seja, a cada imagem o seu índice inicial é acrescido de 5.000. A Figura 8 representa um exemplo desta indexação. O índice utilizado para o recurso *appinfostr* é 4.000.

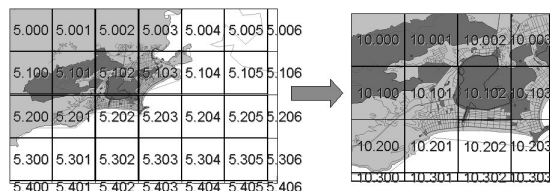


Figura 8: Indexação dos blocos de duas imagens de um mesmo PRC, onde a imagem da esquerda é a mais ampla e a outra é o primeiro nível de zoom.

Transmissão para o Palmtop

Para a transmissão dos arquivos PRC/PDB, gerados no PC, para o Palm, foi utilizado o software de sincronismo que acompanha o Palm (*Palm Desktop*).

Os Comandos no Palm

A utilização de um Palm em operações militares pode dificultar a utilização da caneta na tela, pois o ambiente pode ser hostil não só pela presença de tropas inimigas como pela própria situação do terreno e clima, sem mencionar o desgaste do militar durante a operação. Devido a isso quase todas as operações no Palm podem ser feitas utilizando os seus botões e, sempre que possível, a caneta também. No entanto a medição de distâncias e entrada de dados só são possíveis com toques da caneta na tela. Procedimentos alternativos utilizando os botões podem ser pensados futuramente. A Figura 9 mostra os botões do Palm e suas funções.

Visualização do Mapa Raster

A imagem não é visualizada em toda tela do Palm, pois uma faixa de 10 pixels de altura na parte inferior da tela foi reservada como barra de *status*. Conforme o modo de operação, nesta barra serão exibidas informações referentes a: hora, latitude e longitude em UTM, distâncias, modo de operação e mensagens. Ficando, portanto, uma área de 160 x 150 pixels para a imagem.

Quando é selecionado o modo de zoom (ZOOM) e é pressionado o botão para realizar um *zoom in* verifica-se se o centro da tela está dentro dos limites do mapa seguinte e, se estiver, faz quase todo um processo de inicialização para essa nova imagem, semelhante ao da imagem inicial, desde leitura do recurso *appinfostr* para a imagem, seu po-



Figura 9: As funções dos botões do Palm.

sicionamento, até o desenho de uma área como janela em *offscreen* e desta para a tela, com pequenas variações.

Por ocasião do zoom, o centro da tela atual é o centro da nova tela, se possível, isto é, se não ultrapassar as fronteiras da nova imagem. Para que isso ocorra guarda-se as coordenadas do centro da tela em coordenadas do mundo e depois calcula-se os dados necessários da nova imagem em função destas coordenadas.

O movimento (*panning*) da imagem da tela é feito através dos botões assinalados na Figura 9 e também com a caneta. Com a caneta é feito através das API do PalmOS relativas aos eventos provocados pela caneta quando em contato com a tela. Para suavizar o movimento e não haver pulos de imagens quando houver redesenho da janela em *offscreen* o deslocamento em qualquer direção se dá de, no máximo, 5 em 5 pixels.

No modo de medição de distâncias (DIST), as distâncias que podem ser medidas tem que estar na tela atual. A cada vez que a caneta é encostada à tela, a distância é calculada em relação ao ponto anterior e vai acumulando, em uma variável global, o valor e exibindo-o na barra de *status*. Para inicializar a medição de distância basta pressionar o botão mais à esquerda do Palm, conforme mostra a Figura 9.

Visualização do Mapa Vetorial

Para o desenho do mapa vetorial no PalmOS, um mapa é processada de cada vez. O acesso aos arquivos é necessário sempre que tem que redesenhar o mapa. Alguns conceitos existentes para mapa *raster* são usado no mapa vetorial, mas com algumas alterações. O bloco no caso de desenho de mapa vetorial é do tamanho da janela em *offscreen*. Ele é necessário para definir a localização de um ponto da tela em relação em relação às coordenadas no mundo real. É necessária também as faixas de 160 pixel de largura ou altura em comum entre os blocos, com intuito de evitar os pulos quando cruza suas fronteiras. Este pulo ocorrerá, mas devido ao tempo gasto no redesenho da janela em *offscreen*.

Como o PalmOS só tem como elementos gráficos de desenho o ponto, a linha e o retângulo, os objetos do tipo polígonos foram desenhados ligando seus pontos linha a linha.

Houve a necessidade de fazer um *clipping* dos elementos desenhados para que só fosse desenhada a parte correspondente às dimensões da janela em *offscreen*. Na realidade não foi usado nenhum algoritmo específico. As funções de desenho do PalmOS permite que se desenhe em coordenadas muito além das dimensões da janela em *offscreen*. Só que a parte fora desta foi descartada. Esta área possível de desenhar é cerca de 76 vezes maior do que o tamanho da janela em *offscreen*. O zoom é feito ajustando as coordenadas da imagem dentro destes limites de desenho possível.

Cada vez que o mapa é lido com um nível diferente de zoom ela é inicializada e dividida em blocos. Os blocos servem para acompanhar quando redesenhar o mapa novamente, definindo em que coordenada se fará o redesenho. Esta divisão só é possível com o mapa com estas dimensões, se mudá-lo (zoom) terá que ser feita nova divisão. Diferente do mapa *raster*, essa divisão não gera aumento do banco de dados porque é feita já no Palm e em tempo de execução, sem alteração do banco de dados. A mudança de bloco gera um novo desenho do mapa todo em uma nova posição. O tempo de redesenho torna inviável a utilização dos mapas vetoriais de grandes áreas, como os utilizados em operações militares, devido a isso o trabalho se voltou unicamente aos mapas *raster*. Na seção Resultados são feitas comparações sobre a utilização dos dois formatos.

Aquisição dos Dados do Receptor GPS

A Figura 10 mostra a tela quando em modo GPS e recebendo dados do receptor GPS.

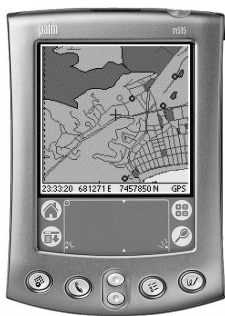


Figura 10: Imagem do aplicativo no modo GPS, recebendo informações.

Para a utilização do receptor GPS conectado ao Palm alguns detalhes dos dois equipamentos devem ser ressaltados: em deslocamentos em operações militares o Palm não pode entrar em *sleep mode* - isso é evitado através das API

do PalmOS; e no protocolo NMEA0183 as coordenadas são fornecidas em graus, minutos e frações de minutos, enquanto os mapas de operações militares terrestres e ribeirinhas são em UTM, portanto, é feita uma conversão delas para UTM.

Neste trabalho, como as informações que nos interessava eram de hora e coordenadas, nos interessava apenas uma sentença do protocolo NMEA0183. Este protocolo é emitido constantemente pelo receptor GPS GARMIN eTrex, a partir do momento que ele é ligado, mesmo que não esteja recebendo sinal de nenhum satélite. O protocolo é emitido no formato ASCII e composto de várias sentenças iniciadas por cadeias de caracteres padronizadas. A sentença que nos interessa é a iniciada por \$GPRMC [4] e contém a hora no meridiano de referência (Greenwich), as coordenadas em graus, minutos e frações de minutos e se a latitude é norte ou sul e a longitude leste ou oeste.

O aplicativo, ao receber os dados do receptor GPS, localiza as coordenadas fornecidas e o coloca como centro da tela, desde que isso seja possível, pois se estiver muito perto das extremidades do mapa o ponto localizado não ficará no centro.

O ponto correspondente às coordenadas recebidas naquele momento é marcado na tela do Palm. À medida que se caminha, a nova posição fornecida é colocada novamente como centro do mapa, redesenhando a tela com um mapa com este ponto central, o que dá uma idéia de movimento do mapa mantendo o ponto central marcado e fixo. Quando chega próximo a uma das bordas do mapa, este para de se deslocar e quem passa a se deslocar é o ponto.

As Observações

A possibilidade de escrever observações no Palm também foi implementada. Para tal foram utilizados recursos do tipo cadeia de caracteres (*string*). Onde ao gravar uma observação cria-se um novo recurso com aquela observação, incrementando o índice do recurso. Como a navegação entre as observações é feita em função dos índices dos recursos, ao apagar uma observação que se encontra no meio de outras é necessário refazer os índices, diminuindo de um todos os índices dos recursos que vem após o que foi apagado. A Figura 11 mostra a tela do modo Observação quando incluindo uma. As observações são escritas com a caneta do Palm usando um formulário em forma de teclado ou o *Graffiti*, ambos já estão previamente instalados no Palm.

2.2 Emprego

A utilização do ambiente proposto neste trabalho propicia:

1. Uma diminuição no tempo de obtenção da posição,



Figura 11: Tela do Palm no modo de observação.

marcação no mapa e comparação com informações existentes em calcos e extratos de planos, já que as coordenadas da posição são obtidas pelo receptor GPS e mostrada na tela do Palm juntamente com o mapa com calcos desenhados nele, bem como de dados escritos sobre a forma de registros de observações. Essa rapidez e precisão das informações será fundamental à operação uma vez que permitirá o emprego das frações certas no momentos e lugares certos.

2. Uma atualização das informações importantes como as referentes ao mapa (não existência de um rio que foi represado, por exemplo) ou à operação (inexistência de um campo minado previsto, por exemplo) através do registro de observações.
3. Um menor desgaste físico dos militares que fazem grandes deslocamentos e, na grande maioria das vezes, não motorizados, pelo emprego de um hardware de peso e dimensões reduzidos que dispensa o transporte de extratos de planos, tabelas, calcos e mapas em papel. Facilitando ainda os deslocamentos em terrenos acidentados e com vegetação abundante ou outros obstáculos.
4. A possibilidade de utilização dos receptores GPS já existentes nas Organizações Militares, desde que permitam conexão com outros equipamentos via porta serial e que utilizem o protocolo NMEA0183 para transmissão dos dados.
5. A facilidade de utilização do hardware, permitindo uma rápida prontificação de militares para sua operação.
6. A aquisição de uma quantidade considerável de hardware devido ao seu baixo custo, propiciando o emprego por um número maior de frações.

3 Resultados

A utilização de mapas vetoriais mostrou-se bastante vantajosa no que se refere a operações de zoom, devido à precisão do novo mapa gerado, mas perde muito em relação ao tempo de redesenho da janela em *offscreen*. A transferência para o Palm também foi mais lenta. Mas o que foi mais marcante e que praticamente inviabilizou sua utilização no Palm, foi o tempo de redesenho. A possibilidade de usar mapas vetoriais mais simples prejudica sua utilização em operações militares onde esses detalhes são importantes. A Figura 12 mostra um gráfico comparativo dos tempos de transmissão do mapa para o Palm e o tempo de desenho para diversos mapas vetoriais e *raster*.

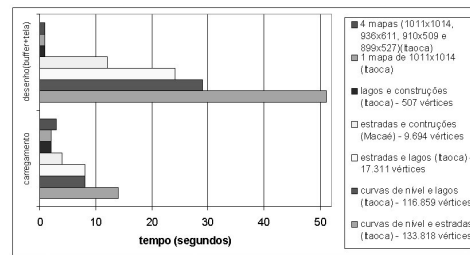


Figura 12: Tempo para carregamento e tempo para desenho do mapa (vetorial x raster).

No início das implementações, foi gerado um arquivo PDB, semelhante ao arquivo contendo o mapa *raster*, só que mais simples, para desenhar o bitmap pixel a pixel, mas foi verificado que a utilização de recursos bitmap em um arquivo PRC permite um desenho mais rápido da imagem do que se for desenhá-la pixel a pixel. A Figura 13 mostra a grande diferença no tempo gasto para o carregamento e o desenho para imagens geradas para desenho pixel a pixel e para desenho do recurso bitmap como um todo.

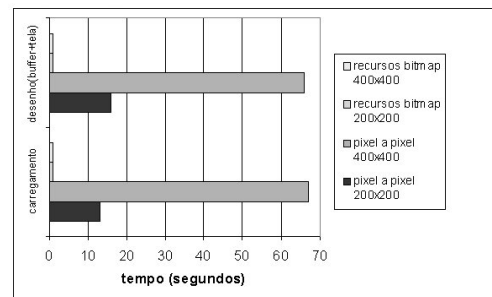


Figura 13: Tempo para carregamento e tempo para desenho da imagem (pixel a pixel x recursos).

Outro problema inicial de utilização de mapas *raster* é o zoom, que se for realizado através das API do PalmOS

ficam truncadas, ou seja, distorcidas, conforme mostram as Figuras 5(a), 5(b), e 5(c). Mas a utilização de outros mapas *raster* para representar o zoom de uma determinada área permitiu uma excelente definição da imagem após o zoom e a possibilidade de utilização também de imagens satélites ou fotografias aéreas (desde que estejam georeferenciadas). Isso sem perda de performance do Palm. A Figura 14 mostra esta utilização de tipos diferentes de imagens (imagens satélite, fotografias aéreas e mapas).

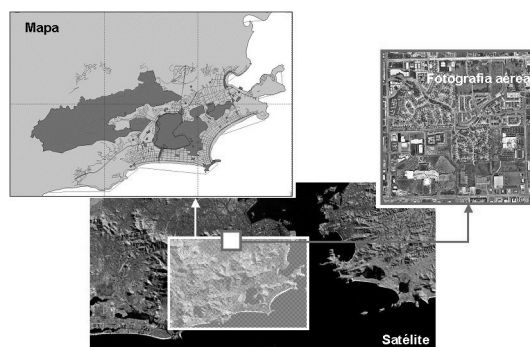


Figura 14: Utilização de diversos tipos de imagens.

Durante os testes foi observada a fragilidade do conector para porta serial do Palm no que se refere a sua fixação para manter esta conexão. Muitos dos testes tiveram que ser feito com o Palm sobre uma mesa para evitar movimentos do conector e a conseqüente perda de conexão com o receptor GPS. Este, no entanto, tem um conector muito bom, com uma fixação excelente. A ativação da conexão com o receptor GPS, o recebimento dos dados e seu tratamento não apresentaram problemas.

4 Conclusão

Englobar todas as possibilidades possíveis para atingir as funcionalidades necessárias ao ambiente da forma mais adequada é um trabalho que necessita um tempo bem mais longo do que o utilizado neste. Mas foi possível tirar algumas conclusões, quais sejam: a utilização do mapa *raster* é mais vantajoso do que mapas vetoriais; a utilização de outros mapas *raster* para representar o zoom de determinadas áreas supre a limitação das API do PalmOS e acrescenta a possibilidade de usar também outros tipos de imagens, tais como imagens satélites e fotografias aéreas; a utilização de mapas *raster* de diferentes fontes, bem como outras imagens, para ressaltar diferentes características de uma mesma área em um mesmo nível de zoom; a possibilidade de utilização de receptores GPS disponíveis nas Organizações Militares; a fragilidade da conexão do Palm com o receptor GPS, em virtude da fixação não robusta do conector do seu cabo serial, necessitando, provavelmente,

de um conector semelhante aos existentes em receptores GPS desenvolvidos especificamente para Palm; a fragilidade do Palm que não possui qualquer resistência a quedas e a água, a despeito do receptor GPS ter baixo custo e, mesmo assim, ser resistente a impactos e cuja resistência à água permite permanecer a um metro de profundidade por trinta minutos. Essa fragilidade do Palm não permite seu emprego incondicional, é necessário ter cuidados que podem tornar seu uso um fardo. A utilização de Palm militarizado é uma saída para este problema, mas eleva seu custo; a utilização deste equipamento permite uma aquisição mais rápida da posição da fração em relação às informações existentes no mapa, permitindo uma maior rapidez no emprego desta no momento e locais adequados; o transporte deste equipamento no lugar de extratos de planos, calcos e mapas, devido a seu baixo peso e pequenas dimensões possibilita um menor desgaste das frações em grandes deslocamentos não motorizados, bem como em terrenos de difícil transposição.

Durante o trabalho surgiram algumas idéias para trabalhos futuros, tais como: implementação de desenho no mapa para a inclusão de dados gráficos, como uma nova estrada ou uma possível área de campo minado, bem como inclusão direto no mapa de símbolos militares padronizados - usando, por exemplo, bitmap com cor de fundo transparente ou representações vetoriais; compressão dos dados na geração dos arquivos de Banco de Dados e durante a manuseio dos mesmos no PDA; estudo sobre desempenho de outras possibilidades de divisão do mapa *raster*, inclusive, utilizando blocos de tamanhos variáveis dependendo da existência de detalhamentos no blocos; aplicativo no PC que leia as informações alteradas no Palm para que elas possam ser utilizadas em operações futuras, como as informações referentes a alterações no terreno; integração de uma rede sem fio interligando os PDAs e equipamentos.

Referências

- [1] Gary Hillerson. *Palm File Format Specification*. Palm Inc., 2001.
- [2] Steve Mann and Ray Rischpater. *Advanced Palm Programming, an Expert's Guide to Creating Killer Application*. Wiley Computer Publishing, 2001.
- [3] Gustavo Moreira Pierre. Ambiente integrado para posicionamento em operações militares. Master's thesis, Pontifícia Universidade Católica do Rio de Janeiro, 2002.
- [4] Neil Rhodes and Julie McKeehan. *Palm Programming: The Developer's Guide*. O'Reilly and Associates, Inc., 1999.