

Načítání a ukládání obrazu

- Předmět: ZPO
 - Zadání: Načítání a ukládání obrazu
 - Řešitelé:
 - Zdeněk Biberle, xbiber00
 - Josef Řídký, xridky00
 - Zdeněk Tretter, xtrett00
 - Datum vypracování: 13. 05. 2015
-

Zadání

Zpracujte funkci pro načítání a ukládání obrazu ve vhodném rastrovém formátu (ne ve formátech .JPG, .BMP, nekomprimovaný .TIF, které jsou k dispozici v knihovně DigLib, zájem naopak je o formáty .GIF, .PCX, .PNG, .TIF s kompresí apod.). Funkci pokud možno zpracujte tak, aby nebyla závislá na jiných knihovnách. Měl by být podporován formát grayscale (256 úrovní šedi - 8 bitů/pixel) a RGB (24 bitů/pixel).

Řešení

Z nabízených formátů si náš tým zvolil formáty GIF a PCX. Rozdělení práce bylo následující:

- GIF - Zdeněk Biberle
- PCX - Josef Řídký, Zdeněk Tretter

U obou formátů bylo cílem vytvoření shared objectu s několika funkcemi s volací konvencí jazyka C, které by měly umožnit nahrávání a ukládání obou formátů dle zadání.

Knihovna libzpogif.so

Tato knihovna slouží k nahrávání a ukládání obrázků ve formátu GIF verze 89a. Knihovna dává uživateli volnost co se týče uložení obrázků v paměti. Toho je dosaženo parametrizací velikosti kroku mezi jednotlivými pixely řádků obrazu a mezi jednotlivými řádky. Je dokonce možné použít i zápornou velikost kroku.

Knihovna také dává uživateli volbu metody alokace obrazu při nahrávání. Toho je dosaženo předáním alokační a dealokační funkce do funkce nahrávací. Alokační funkce v tomto případě určuje právě i dříve zmíněné velikosti kroků.

Knihovna rozeznává všechny vlastnosti formátu GIF verze 89a. Knihovna ovšem nepodporuje nahrávání ani ukládání animovaných obrázků. Rozšíření Comment Extension, Plain Text Extension a Application Extension jsou ignorována.

Knihovna má následující rozhraní s volací konvencí jazyka C:

```
zpogif_error zpogif_save(FILE* f,
    zpogif_format format,
    const void* image,
    uint16_t width,
    uint16_t height,
    ptrdiff_t pixel_stride,
    ptrdiff_t row_stride);
```

```

zpogif_error zpogif_load(FILE* f,
    zpogif_format format,
    void** image_out,
    uint16_t* width_out,
    uint16_t* height_out,
    ptrdiff_t* pixel_stride_out,
    ptrdiff_t* row_stride_out,
    void* (*allocator)(zpogif_format, uint16_t, uint16_t, ptrdiff_t*,
    ptrdiff_t*, void*),
    void (*deallocator)(zpogif_format, uint16_t, uint16_t, void*, void*),
    void* allocator_data);

```

Jak je z názvu zjevné, funkce zpogif_save slouží k ukládání obrázků a funkce zpogif_load slouží k nahrávání obrázků.

Hlavičky obou funkcí určené pro uživatele knihovny se nachází v hlavičkovém souboru zpogif.h.

Funkce zpogif_save

Tato funkce přečte obrázek uložený v paměti a uloží jej do dodaného souboru.

Parametry funkce jsou následující:

- f - soubor, do kterého bude obrázek zapsán
- format - Formát pixelů obrázku, tento parametr má dvě hodnoty:
 - ZPOGIF_RGB - Každý pixel je uložen jako trojice po sobě jdoucích bajtů s úrovněmi červené, zelené a modré barvy.
 - ZPOGIF_GRAYSCALE - Každý pixel je uložen jako jeden bajt s hodnotou jasu.
- image - Ukazatel na první pixel obrazových dat. Adresy dalších pixelů se odvíjí od této hodnoty na základě následujících čtyř parametrů.
- width - Šířka obrázku v pixelech.
- height - Výška obrázku v pixelech.
- pixel_stride - Velikost kroku mezi pixely na řádků. Při procházení obrázku bude funkce zpogif_save používat tento parametr k posunu na další pixel na řádku.
- row_stride - Funguje podobně jako parametr pixel_stride, ovšem určuje velikost kroku mezi řádky.

Funkce vrací hodnotu typu zpogif_error, která označuje, zda došlo při ukládání k nějaké chybě.

Funkce zpogif_load

Tato funkce přečte obrázek ze souboru a uloží jej do paměti, kterou alokuje pomocí dodaného alokátoru.

Parametry funkce jsou následující:

- f - soubor, ze kterého bude obrázek přečten
- format - Formát pixelů obrázku, tento parametr má dvě hodnoty:
 - ZPOGIF_RGB - Každý pixel bude uložen jako trojice po sobě jdoucích bajtů s úrovněmi červené, zelené a modré barvy.
 - ZPOGIF_GRAYSCALE - Každý pixel bude uložen jako jeden bajt s hodnotou jasu.
- image_out - Ukazatel na ukazatel na první pixel obrázku. Tento parametr slouží jako výstup.

- `width_out` - Ukazatel na šírku obrázku v pixelech. Tato hodnota bude funkcí naplněna po přečtení souboru.
- `height_out` - Ukazatel na výšku obrázku v pixelech. Tato hodnota bude funkcí naplněna po přečtení souboru.
- `pixel_stride_out` - Ukazatel na krok mezi jednotlivými pixely na řádku. Tato hodnota bude naplněna hodnotou, kterou během čtení obrázku dodá alokátor.
- `row_stride_out` - Ukazatel na krok mezi jednotlivými řádky obrázku. Tato hodnota bude naplněna hodnotou, kterou během čtení obrázku dodá alokátor.
- `allocator` - Ukazatel na funkci, která slouží pro alokaci nahrávaného obrázku.
- `deallocator` - Ukazatel na funkci, která slouží pro dealokaci nahrávaného obrázku.
- `allocator_data` - Ukazatel na libovolná uživatelská data, která budou předána do alokační a dealokační funkce.

Funkce vrací hodnotu typu `zpogif_error`, která označuje, zda došlo při nahrávání k nějaké chybě.

Alokace a dealokace obrázků

Jak již bylo zmíněno, tak při nahrávání obrázku ze souboru je k alokaci obrázku v paměti použita uživatelem dodaná alokační funkce. Uživatel by taktéž měl dodat funkci dealokační, aby bylo možné uvolnit alokovanou paměť v případě chyby během nahrávání obrázku.

Alokační funkce má šest parametrů v následujícím pořadí:

- Formát obrázku - `ZPOGIF_RGB` nebo `ZPOGIF_GRAYSCALE`.
- Šířka obrázku v pixelech.
- Výška obrázku v pixelech.
- Výstupní ukazatel na krok mezi pixely na řádku.
- Výstupní ukazatel na krok mezi řádky obrázku.
- Ukazatel na libovolná uživatelská data.

Alokační funkce musí vrátit ukazatel na první pixel obrázku nebo `NULL`, pokud se alokace nezdařila.

Dealokační funkce má tyto parametry:

- Formát obrázku - `ZPOGIF_RGB` nebo `ZPOGIF_GRAYSCALE`.
- Šířka obrázku v pixelech.
- Výška obrázku v pixelech.
- Ukazatel na první pixel obrázku.
- Ukazatel na libovolná uživatelská data.

Princip výpočtu adres pixelů v obrázku

Jak již bylo zmíněno, tak tato knihovna dává uživateli možnost zvolit si způsob uložení obrázku v paměti. Toho je docíleno především pomocí parametrů `pixel_stride` a `row_stride`.

Funkce `zpogif_save` a `zpogif_load` používají tyto parametry k výpočtu posunu jednotlivých pixelů v paměti relativně k adrese prvního pixelu. Tento výpočet probíhá následovně:

$\text{offset} = y * \text{row_stride} + x * \text{pixel_stride}$

Tento způsob definice adresování má hlavní výhodu v tom, že uživatel může zaručit zarovnání pixelů a řádků nezávisle na velikost obrázku. Představme si například situaci, kdy máme RGB obrázek a chceme, aby každý pixel byl zarovnán na čtyři bajty řádek na šestnáct bajtů. V tom případě zvolíme `pixel_stride` rovno čtyřem a `row_stride` násobek šestnácti.

C++ rozhraní

Knihovna poskytuje i C++ rozhraní, které je velice podobné popsanému C rozhraní, ovšem s několika drobnými rozdíly:

- Funkce se nachází ve jmenném prostoru `zpogif`.
- Funkce se jmenují `save` a `load`.
- Obě dvě funkce mohou mít jako vstupní a výstupní soubor typ `FILE*` nebo `std::istream`, resp. `std::ostream`.
- Ohlašování chyb chybovými kódy je nahrazeno výjimkami.
- Místo ukazatelů na alokační a dealokační funkce je použita šablona `std::function<...>`.
- Funkce `load` již nepřijímá ukazatel na uživatelská data pro alokační a dealokační funkci.

Toto C++ rozhraní se nachází v hlavičkovém souboru `zpogif.hpp`.

Knihovna `libzpopcx.so`

Tato knihovna slouží k nahrávání a ukládání TrueColor RGB a šedotónových obrázků ve formátu PCX ve verzi 3.

Knihovna má následující rozhraní s volací konvencí jazyka C:

```
int saveToPCX(char *out);  
int loadFromPCX(char *in);
```

Jak je z názvu zjevné, funkce `saveToPCX` slouží k ukládání obrázků a funkce `loadFromPCX` slouží k nahrávání obrázků. Funkce `loadFromPCX` automaticky podle informací v hlavičce PCX souboru rozpozná, o jaký typ obrazu (šedotónový, TrueColor RGB) se jedná. Funkce `saveToPCX` poté na základě této detekce ukládá soubor v potřebné barevné verzi.

Hlavičky obou funkcí určené pro uživatele knihovny se nachází v hlavičkovém souboru `zpopcx.h`.

Knihovna pracuje pouze s obrázky, které jsou komprimovány RLE kompresí.

Pro interní reprezentaci obrázku je využita třída `BMP`, která obsahuje metody pro načtení a uložení obrázku. Tato třída si uchovává potřebné informace z hlavičky načítaného obrázku jako například jeho šířku, výšku, počet bitů na pixel, počet bitových rovin apod. Tyto informace jsou poté při ukládání obrázku do souboru znovu využity.

Funkce `saveToPCX`

Tato funkce převede obrázek z interní reprezentace do požadovaného souboru. Dle typu souboru (šedotónový, TrueColor RGB) jsou následně volány funkce `saveToGrayscalePCX` a `saveToTrueColorPCX`, které zajišťují samotné uložení a komprimaci interní reprezentace obrázku.

Parametry funkce jsou následující:

- `out` - soubor, do kterého bude obrázek uložen

Funkce vrátí hodnotu typu `int`, která označuje, zda došlo při ukládání k nějaké chybě.

Funkce `loadFromPCX`

Tato funkce přečte obrázek ze souboru a převede jej do interní reprezentace.

Parametry funkce jsou následující:

- `in` - soubor, ze kterého bude obrázek přečten

Funkce vrátí hodnotu typu `int`, která označuje, zda došlo při nahrávání k nějaké chybě.

Obrázky jsou ukládány s využitím komprimace algoritmem RLE.

Použité zdroje

GRAPHICS INTERCHANGE FORMAT(sm). *World Wide Web Consortium*. [online]. 31.7.1990 [cit. 2015-05-13]. Dostupné z: <http://www.w3.org/Graphics/GIF/spec-gif89a.txt>

PCX Header Format [online]. [cit. 2015-05-13]. Dostupné z: http://www.fastgraph.com/help/pcx_header_format.html

Grafický formát PCX - výlet do historie PC. *ROOT.cz*. [online]. 11.6.2006 [cit. 2015-05-13]. Dostupné z: <http://www.root.cz/clanky/graficky-format-pcx-vylet-do-historie-pc/>

What's In A GIF. *Project Labs*. [online]. 24.1.2005 [cit. 2015-05-13]. Dostupné z: <http://www.matthewflickinger.com/lab/whatsinagif/index.html>