

FACE DETECTION IN CAMERA IMAGE ON A MOBILE PHONE

Martin Tureček

Master Degree Programme (2), FIT BUT

E-mail: xturec03@stud.fit.vutbr.cz

Supervised by: Adam Herout

E-mail: herout@fit.vutbr.cz

ABSTRACT

This paper presents application for a real-time face detection on Windows Mobile platform. The application uses DirectShow to construct a filter graph which captures a video from a camera device. For a face detection itself, custom transform filter was developed. It uses classifier trained by WaldBoost algorithm and Haar or Local Rank Differences features.

1. ÚVOD

Předmětem této práce je experimentovat s možnostmi dnešních mobilních zařízení ve výpočetně náročných aplikacích. Pro tyto účely byla vytvořena aplikace, která detekuje obličej v obraze z kamery mobilního zařízení. Konkrétně byla zvolena platforma Windows Mobile (dále jen WM).

2. ROZBOR

Vývoj aplikace probíhal ve dvou krocích, které na sebe bezprostředně navazovaly. V prvním z nich bylo třeba analyzovat možnosti vývoje na platformě WM a zejména se zaměřit na práci s kamerou, jež je dostupná na drtivé většině zařízení. Dalším krokem bylo seznámit se s metodami rychlé detekce obličej v obraze a zjistit, která kombinace algoritmů a příznaků je nejvhodnější pro použití v případě dostupnosti pouze velmi malého výpočetního výkonu.

2.1. PRÁCE S KAMEROU

Prakticky jedinou použitelnou alternativou pro práci s kamerou na platformě WM je multimediální framework DirectShow (dále DS), jenž je dostupný nejen na desktopové verzi systému Windows, ale i na té mobilní, a měl by tak zaručovat bezproblémový chod aplikace na všech mobilních zařízeních s WM.

2.2. DETEKCE OBLIČEJE

Detekce obecně je úloha, která má za úkol zjistit, zda jsou v obraze objekty zájmu (v tomto případě obličej). Jedná se o složitou úlohu zejména kvůli velké variabilitě objektů. Výraz

obličej, jeho natočení, velikost, nehomogenní osvětlení, či různá zkreslení kamery jako jsou šum nebo rozmazání jsou jen některými z mnoha faktorů, které detekci velmi znesnadňují.

Aplikace bude používat jen příznakově orientované systémy, které jsou oproti pixelově orientovaným systémům rychlejší a flexibilnější. Existuje několik druhů příznaků, které se mimo jiné liší vhodností pro různé cílové platformy. Aplikace v současné době využívá klasifikátory s LRD [1][2] nebo Haarovými příznaky, jenž operují nad integrálním obrazem [3]. Klasifikátory byly trénovány metodou WaldBoost [4], která je kombinací metody AdaBoost [3] a SPRT.

3. REALIZACE

Aplikace tedy využívá framework DS a celá komplexní úloha tak mohla být rozdělena do dílčích kroků, které se nazývají filtry. Ty jsou sdruženy do grafu filtrů, který se skládá z filtrů, které jsou již součástí frameworku DS, a z filtrů vlastních. Prvním filtrem je filtr pro zachytávání obrazu z kamery mobilního telefonu. Jedná se o standardní filtr, který by měl být snadno konfigurovatelný. To se ale nakonec nepotvrdilo. Jiné než výchozí rozlišení kamery na některých zařízeních vůbec nešlo nastavit (HTC HD). U dalšího zařízení (HTC HD2) nastavit rozlišení možné bylo, obraz byl ovšem otočený o 90 stupňů. Pokoušel jsem se tento zásadní nedostatek řešit i s technickou podporou firmy HTC, ta však nedokázala odpovědět a odkázala mě pouze na neoficiální fórum, kde podobný problém řešilo neúspěšně hned několik vývojářů.

Výsledkem snahy o správnou orientaci obrazu byl vlastní transformační filtr, který umí orientaci obrazu upravit. Tento filtr pak slouží jako zdroj dalšího vlastního filtru, který provádí samotnou detekci obličejů v obraze.

Filtr pro detekci zatím přijímá pouze obraz v barevném prostoru YV12. V první fázi je nutno vytvořit ze vstupního obrazu pyramidu obrazů, jejíž úrovně obsahují podvzorkovaný původní obraz (používá se metoda nearest neighbour, která je svou kvalitou dostatečná a zároveň nenáročná na výkon). To zajistí správnou detekci obličejů v různých vzdálenostech od objektu.

Samotná detekce pomocí klasifikátoru pak probíhá tak, že pro každý obraz pyramidy je bod po bodu přikládáno pod-okno o konstantní velikosti (typicky 24x24 obrazových bodů). Toto okno je poté klasifikováno pomocí silného klasifikátoru, který se skládá z velkého počtu slabých klasifikátorů. Výhodou tohoto přístupu je velmi rychlé odmítnutí pod-oken, jejichž obsahem je například jen pozadí, zatímco cenný výpočetní výkon je věnován slibnějším pod-oknům.

V současné době jsou implementovány dva klasifikátory, které se liší používanými příznaky. První klasifikátor používá LRD příznaky, zatímco druhý klasifikátor Haarovy příznaky (ty pracují s integrálním obrazem). Data natrénovaných klasifikátorů jsem obdržel v souboru XML. Ovšem vzhledem k poměrně problematickému parsování tohoto typu souboru na platformě WM jsem převedl tento soubor do C++ hlavičkového souboru, který je součástí filtru.

4. VÝSLEDKY

Současná aplikace zatím umožňuje jen subjektivní srovnání klasifikátorů natrénovaných metodou WaldBoost za použití LRD nebo Haarových příznaků. Aby bylo dosaženo co nejvyšší rychlosti detekce, bylo experimentováno se zjednodušováním výpočtu na úkor kvality detekce – např. omezení počtu úrovní pyramidy obrazů, či přikládání pod-okna jen každé dva pixely. Tato omezení přinesla významné zvýšení výkonu a kvalitu detekce nijak významně neovlivnily.

Aplikace byla doposud testována na dvou zařízeních od firmy HTC (s takty procesoru 528 MHz a 1 Ghz). Detekce pomocí LRD příznaků je subjektivně plynulá na nejrychlejších zařízeních (cca 25 fps) do rozlišení 576x480. Pro vyšší rozlišení je již znatelný pokles snímků za vteřinu. I maximální rozlišení (800x480) je ovšem pořád dostatečně rychlé (cca 8 fps). Současná implementace detekce pomocí Haarových příznaků je zatím bohužel velmi pomalá i pro malá rozlišení (120x160 – cca 0,5 fps). Budoucím předmětem této práce bude tedy zjistit, co je příčinou takto pomalé detekce, případně tuto chybu odstranit.

5. ZÁVĚR

Samotná aplikace umožnila srovnání výkonu běžných stolních počítačů s nejvýkonnějšími mobilními zařízeními. Toto srovnání (zatím jen subjektivní) ukázalo, že výkon dnešních mobilních telefonů je již dostatečný i pro výpočetně velmi náročné aplikace, jakými je například detekce obličeji v obraze. V současnosti nejrychlejší zařízení (např. HD2) tuto úlohu zvládají v reálném čase i pro relativně vysoká rozlišení.

Ačkoliv byl použit framework DS, který by měl zaručovat kompatibilitu aplikace na všech zařízeních, tak se během vývoje aplikace ukázalo, že většina výrobců vyvíjí pro své fotoaparátové aplikace vlastní ovladače, k nimž není přístupný žádný SDK a navíc jsou nedokumentované. Spolupráci kamery s frameworkem DS pak odbývají, což ústí v nekompatibilitu jinak bezproblémově fungujících implementací na některých zařízeních (např. rotace obrazu o 90° u HTC HD2 apod.). Platforma WM mě tedy v tomto ohledu poměrně zklamala a jen potvrdila, že je častým cílem kritiky oprávněně.

Poděkování: Tato práce vznikla částečně za podpory grantu VUT FIT, FIT-S-10-2 a specifického výzkumu MSM0021630528.

LITERATURA

- [1] Zemčík, P., Hradiš, M., Herout, A.: Local Rank Differences – Novel Features for Image Processing, Proceedings of SCCG 2007, Budmeřice, 2007
- [2] Polok L., Herout A., Zemčík P., Hradiš M., Juránek R., Jošth R.: „Local Rank Differences“ Image Feature Implemented on GPU, 2008 Proceedings of ACIVS 2008
- [3] Viola, P., Jones, M.: Robust Real Time Object Detection, Statistical and Computational Theoresis of Vision, Vancouver, Canada, 2001
- [4] Šochman, J., Matas, J.: WaldBoost – Learning for Time Constrained Sequential Detection, Center For Machine Perception, Dept. of Cybernetics, Faculty of Elec. Eng., CVUT, Prague, 2005