

GAZE ESTIMATION IN CONTROLLING COMPUTER GAME

Jaromír Vaňhara

Bachelor Degree Programme (3), FIT BUT

E-mail: xvanha01@stud.fit.vutbr.cz

Supervised by: Michal Hradiš

E-mail: ihradis@fit.vutbr.cz

ABSTRACT

The purpose of this paper is to show possible way how to control computer with human eyes. The system presented here should extend communication possibilities, e.g. for games, for disabled people, etc. Whole problem of human eyes gaze estimation can be divided into three parts – detecting eyes, detecting pupil or iris and the eye gaze estimation itself.

1 ÚVOD

Na rozdíl od jiných oblastí výpočetní techniky se způsoby komunikace počítače s člověkem příliš nerozvíjí. V současné době se pro ovládání počítače běžným uživatelem používá hlavně klávesnice a myš, tedy způsoby používané již dlouhou dobu. Hledání nových cest, jak počítače ovládat může vést k dalšímu velkému rozvoji výpočetní techniky. Další výstupy mohou být užitečné i v kratším horizontu, jako například zjednodušení přístupu k počítačům pro zdravotně postižené.

2 ROZBOR

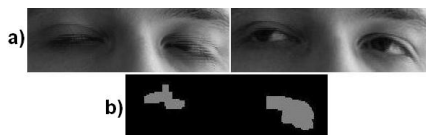
Cílem práce je tedy vytvořit knihovnu, která bude implementovat funkce na zjištění směru pohledu. Z důvodů popsaných dále, bude tato knihovna jako vstup přijímat dva po sobě následující obrazy z kamery snímající obličej člověka nebo jeho určitou část. Jejím výstupem bude informace o směru pohledu tohoto člověka.

Celý problém se dá rozdělit do tří částí: nalezení očí v obraze, nalezení duhovky nebo zorničky v oblasti očí a určení směru pohledu.

3 NALEZENÍ OČÍ V OBRAZE

Jako základ pro nalezení očí v obraze jsem použil metodu, kterou navrhli Chau a Betke [1]. Je to způsob využívající mrkání člověka, ať už samovolného nebo úmyslného. Tato metoda byla zvolena z důvodu její jednoduchosti a správné funkčnosti i při nízkém rozlišení kamery. Přímou v sobě také implementuje možnost použití detekovaného mrknutí, jako jednoho z ovládacích prvků případné hry ovládané očima.

Principem této metody je zjištění rozdílu dvou po sobě následujících snímků získaných z kamery snímající obličej člověka. Po vyprahování rozdílu získáme binární obraz ukazující místa, kde došlo mezi těmito snímky k pohybu. Pro odstranění šumu způsobeném nedokonalostí kamery se po vyprahování na obraze provede dilatace a eroze. Případné mrknutí, tedy pohyb, se pak na binárním obraze zřetelně projeví, viz obr. 1.



Obrázek 1: Příklad mrknutí (a) a vytvořeného binárního obrazu (b)

Dále se provede detekce spojitých oblastí. Samozřejmě ne všechny nalezené spojité oblasti jsou oči. Na základě tvaru a pozice očí na lidském obličejí jsem stanovil podmínky, po jejich splnění považuji za oči:

- dvě spojité oblasti
- mající určitou minimální velikost
- jsou v přibližně stejné výšce
- musejí být ve vzálenosti větší, než je jejich šířka

Jsou-li tyto podmínky splněny, jsou dané oblasti považovány za oči. Pokud takto nalezneme spojité oblasti i po prvotní fázi inicializace a překrývají se s očima nalezenými před tím, můžeme to považovat za stav, kdy člověk mrknul. Na rozdíl od metody popsané autory Chau a Betke [1] nepoužívám pro sledování již nalezených očí template matching, ale metodu optical flow. Tím se dají přesněji koriovat drobné pohyby.

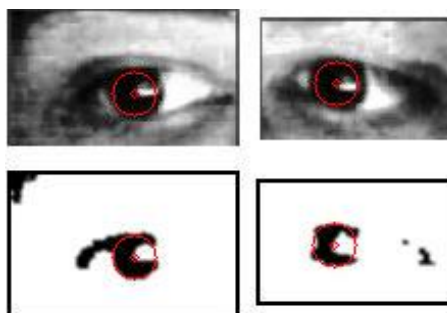
4 NALEZENÍ DUHOVKY NEBO ZORNIČKY V OBLASTI OČÍ

Postupem popsaném v předchozí části jsem tedy našel dvě spojité oblasti, které považuji za oblasti očí. Detekce duhovky, případně zorničky, je vlastně nalezení kruhové spojité oblasti. K detekci používám Houghovu transformaci. Před touto transformací vždy oblast, kde hledám, normalizuji a pak postupně prahuji se zvyšujícím se prahem. Ve chvíli, kdy naleznu kruh v dané oblasti, cyklus, ve kterém prahuji, ukončím. Stejný postup provedu i pro druhou oblast.

Výhodou také je, že postupným prahováním každého oka zvlášť, dokáží eliminovat problém, kdy jsou oči různě nasvíceny, třeba při bočním světle. Příklad takto nalezených duhovek je na obr. 2.

5 URČENÍ SMĚRU POHLEDU

K určení směru pohledu potřebuji získat polohu očí, které se dívají do významných bodů, například do rohů obrazovky. Uživatel tedy bude požádán o to, aby se do těchto míst díval. V



Obrázek 2: Normalizovaný obraz okolí očí před a po vyprahování. Nalezené duhovky jsou zvýrazněny.

každém bodě se zaznamená přesná poloha očí. Při vlastním používání se místo, kam se člověk dívá, získá lineární interpolací. Pro fungování tohoto postupu je ale důležité, aby se osoba, kromě pohybu očí a mrkání, nehýbala.

6 VÝSLEDKY

V současné chvíli jsou implementovány první dva body – nalezení očí v obraze a nalezení duhovky. Pro implementaci je využita knihovna OpenCV [2] společnosti Intel a implementační jazyk je C++.

Pro vyhledání očí v obraze jsem zjistil několik důležitých poznatků. Za prvé je třeba, aby na pozadí nebyl žádný pohyb, nebo aby kamera snímala pouze obličej člověka, tedy bez pozadí. Dále je třeba, aby se člověk, kromě mrkání nebo pohybu očí, nehýbal. Pokud jsou splněny tyto předpoklady, tak jsou oči s dostatečnou přesností detekovány.

Popsaný způsob detekce duhovky funguje dobře při různém osvětlení. Postup zatím nebyl testován na dostatečném vzorku, ale z dosavadních testů vyplývá, že metoda funguje pro různě barevné duhovky. Je tady ale ještě prostor k zpřesnění detekce.

Zdá se také, že kamera s nízkým rozlišením nebude dostatečně přesná ke správnému určení směru pohledu. Nabízí se použití HD kamery, kde je dostatečný rozdíl v poloze očí sledujících levou a pravou část obrazovky, řádově v desítkách pixelů.

7 ZÁVĚR

Po dokončení poslední části programu, plánuji vytvořit hru, která by se dala pomocí očí ovládat. Otestování funkčnosti je potřeba provést na větším vzorku lidí s různě barevnými duhovkami. Na základě tohoto testování se dají jednotlivé části dále zpřesnit.

REFERENCE

- [1] Chau M., Betke M.: Real Time Eye Tracking and Blink Detection with USB Cameras, Boston University Computer Science Technical Report No. 2005-12.
- [2] Intel OpenCV documentation, <http://opencvlibrary.sourceforge.net>