

Vysoké učení technické v Brně
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií



Stabilizace video-sekvence

Ročníkový projekt

Abstrakt

Ročníkový projekt řeší problém stabilizace videozáznamu pořízeného kamerou. K řešení je použita gradientní metoda. Na základě nalezení gradientů v po sobě dvou následujících snímcích pomocí Sobelova operátoru, je pomocí několika iteračních kroků určen a dále zpřesňován optický tok. Optický tok udává rychlost a směr, jakým se jednotlivé objekty vůči snímacímu zařízení v obraze pohybují. Na základě sumarizace optického toku jednotlivých částí obrazu, můžeme určit, zda je scéna v klidu, nebo zda objekty na ní zachycené mění svoji polohu.

Abstract

The project solve problem about stabilization of video sekvence, which is made by camera. We use the gradient method to solve this problem. In the first step we found the gradient intensity in image using the Sobel operator. We found basic optic flow and in some other iteration steps computing new value of optic flow. The optic flow sets rate and direction, what every part in picture move. On the basis summarization optical flow single part picture, we can determine, whether is scene quiet, or whether objects on her captured turns its position.

Klíčová slova

stabilizace videa, optický tok, FFT, Sobel, konvoluce

Key words

stabilization of video, optic flow, fast fouiour transform, Sobel, convolution

Prohlášení

Prohlašuji, že jsem tento ročníkový projekt vypracoval samostatně pod vedením Ing. Vítězslava Berana.

Další informace mi poskytli.....

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Martin Beran

OBSAH

| | |
|---|----|
| 1. Úvod | 6 |
| 2. Videozáznam | 6 |
| 2.1 Záznam videosignálu | 6 |
| 2.2 Obraz a jeho reprezentace | 7 |
| 2.3 Digitalizace obrazu | 7 |
| 2.3.1 Kvantování | 7 |
| 2.3.2 Vzorkování | 8 |
| 2.3.3 Fourierův obraz | 8 |
| 3. Optický tok | 10 |
| 3.1 Metody výpočtu optického toku | 11 |
| 3.1.1 Korelační metoda | 11 |
| 3.1.2 Gradientní metoda | 12 |
| 3.2 Výpočet gradientu | 15 |
| 3.3 Horn a Schunck algoritmus | 17 |
| 4. Závěr | 20 |
| 4.1 Zhodnocení vlastní realizace | 20 |
| 4.2 Možnosti dalšího rozvoje | 20 |

1. Úvod

V době velkého elektronického pokroku a současné miniaturizaci je třeba nestabilitu snímacího zařízení eliminovat. Tento problém řeší projekt **stabilizace video-sekvence**. Cílem projektu je navrhnout algoritmus, jehož aplikací na nestabilní videozáznam, získáme stabilizovaný obraz.

Problém je řešen v několika fázích:

V první fázi je třeba si uvědomit, že s videozáznam nemůžeme pracovat jako s celkem, ale musíme zpracovávat jednotlivé obrázky (půlsnímky) samostatně.

V druhé fázi se zaměřuje na hledání zajímavých bodů v obraze a jejich porovnání s dalšími snímky.

V poslední fázi zajistíme plynulý přechod mezi jednotlivými snímky obrazu.

2. Videozáznam

S videozáznamem se můžeme setkat ve dvou verzích. V první řadě jako analogový signál (TV vysílání, videorekordér, atd.) nebo ve stále se rozšiřující digitální verzi (videokamery atd.). Analogový signál se šíří jako frekvenční spektrum, jehož amplituda určuje výsledný obraz. Digitální video je definováno jako střídající se úrovně logických jedniček a nul. Tyto logické úrovně nesou informaci o velikosti jednoho bitu. V další části už budeme pracovat pouze se signálem v digitálním tvaru.

2.1 Záznam videosignálu

Veškeré děje a situace, které se okolo nás odehrávají, mají spojitý charakter. Tuto vlastnost však nejsme schopni zachytit pomocí zařízení na tvorbu videa. Pokud ale skutečný obraz zaznamenáváme sice diskrétně, ale s dostatečně velkou frekvencí, jsme schopni dosáhnout záznamu, podobajícímu se reálnému obrazu.

Tohoto principu využívají videokamery, které vstupní spojitý signál zaznamenávají jako posloupnost diskrétních obrazů, které při následném promítání dávají opět spojitý dojem. Diskrétní obraz si můžeme představit jako matici $m \times n$ pixelů, kde m je počet řádků matice a n počet sloupců. Vzorkovací frekvence záznamu je 24 fps (frame per second). Ve skutečnosti je ale dvojnásobná, neboť v jednom snímku jsou zaznamenány pouze liché řádky matice a v následujícím naopak sudé.

2.2 Obraz a jeho reprezentace

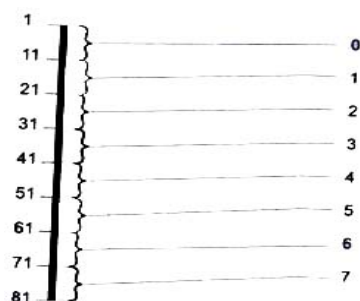
Na definici pojmu obraz, se díváme každý jinak. Pro někoho jde o fotografii, obrázek na papíře, odraz ve vodní hladině, nebo třeba obraz vytvořený na sítnici oka. Matematicky však můžeme obraz definovat jako spojitou funkci dvou proměnných $z=f(x,y)$. Definičním oborem této funkce je spojitý interval $I \subset \mathbb{R}^2$, x a y jsou souřadnice bodu ve 2D, v němž funkce nabývá nějaké hodnoty ze svého definičního oboru. Téměř nikdy však nepracujeme se spojitým definičním oborem funkce. Daleko častěji pracujeme v bodovém rastru. Základním elementem obrazu je *pixel*, z anglického *picture element*.

2.3 Digitalizace obrazu

Pro získání digitálního obrazu je třeba přejít od spojité funkce $f(x,y)$ k diskrétní funkci $I(x,y)$, a to jak v definičním oboru funkce $f(x,y)$, tak v jejím oboru hodnot. Tento proces se nazývá *digitalizace* a odehrává se ve dvou krocích. Jedná se o *kvantování* a *vzorkování*.

2.3.1 Kvantování

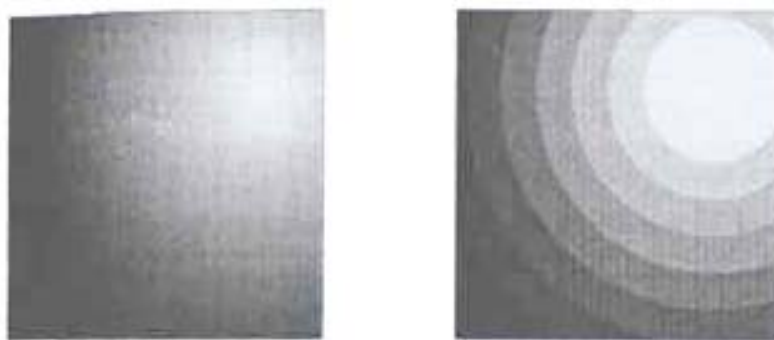
Kvantování diskretizuje obor hodnot obrazové funkce. Obor hodnot se rozdělí na intervaly, jimž je přidělena jediná zástupná hodnota, viz obrázek 2.3-1.



Obr. 2.3-1: Kvantování

Podle toho, zda je kvantovací veličina konstantní či nikoli, hovoříme o *uniformním* nebo *neuniformním* kvantování. Hodnota kvantovací veličiny je volena různými způsoby. Buď jako průměr hodnot z celého intervalu, nebo jako vážený průměr, medián atd.

Aplikací kvantování na obraz s malou změnou gradientu, má za následek vzniku tzv. *kvantizační chyby*, která se projevuje náhlým skokem v hodnotě barev, viz obrázek 2.3-2. Tato chyba se dá odstranit aplikací neuniformního kvantování.

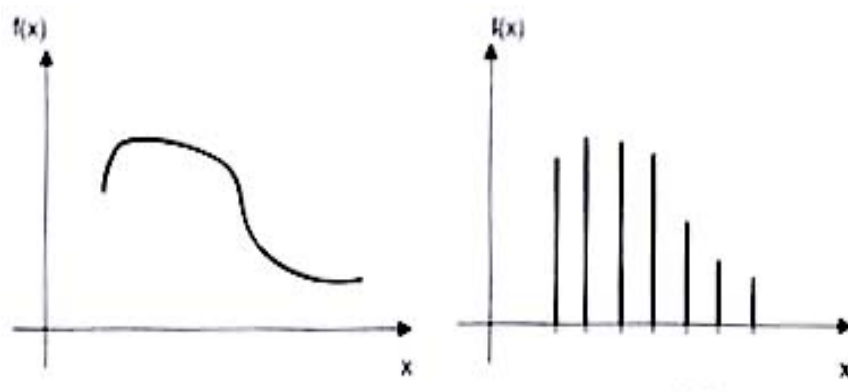


2.3-2: Původní obraz (vlevo), chyba vzniklá kvantováním (vpravo)

2.3.2 Vzorkování

Jde o metodu, kdy jsou z původní spojité funkce $f(x,y)$ odebírány vzorky v konstantních intervalech. Čím je vzdálenost dvou sousedních vzorků menší, tím je původní signál reprezentován přesněji. Vzdálenost dvou vzorků označme Δx . *Vzorkovací frekvenci* pak rozumíme hodnotu

$$f_s = \frac{1}{\Delta x} \text{ (viz obrázek 2.3-3).}$$



2.3-3: Vzorkování

Čím bude vzorkovací frekvence vyšší, tím bude diskretní obraz přesnější, avšak vyšší frekvence znamená vyšší paměťové nároky. Otázkou zůstává, zda jsme schopni přesně reprezentovat spojitý signál v diskretní mřížce. K zodpovězení této otázky nám může pomoci reprezentace funkce jako *Fourierův obraz*.

2.3.3 Fourierův obraz

Obraz je reprezentován dvěma způsoby. Pokud se jedná o diskretní obraz reprezentovaný maticí pixelů, jde o obraz v *prostorové* nebo také *časové oblasti*. *Fourierův obraz* je reprezentací našeho obrazu v tzv. *frekvenční oblasti*. Tento obraz je složen z nekonečně mnoha sinusovek s různou amplitudou a vzájemným fázovým posunutím. Přejít, kterým získáme Fourierův obraz z obrazu

diskrétního, se nazývá *dopředná Fourierova transformace*. Původní obraz získáme *zpětnou Fourierovou transformací*.

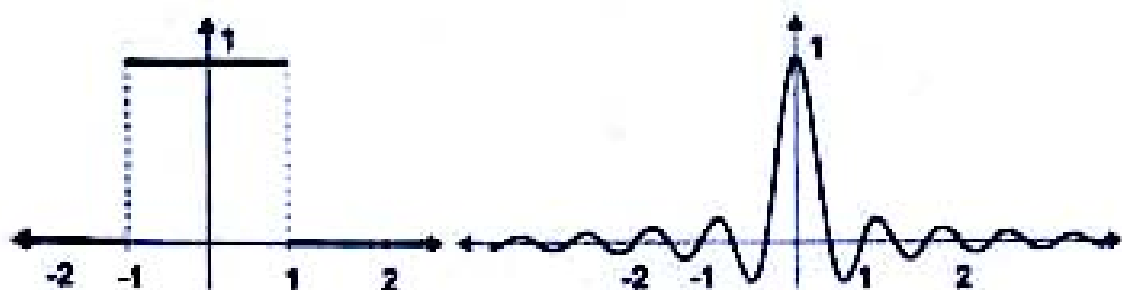
Fourierův obraz jednorozměrné funkce $f(x)$ definované na spojitém definičním oboru, rozumíme funkci $F(u)$, kterou získáme integrací

$$F(u) = \int_{-\infty}^{+\infty} f(x) e^{-i 2 \pi u x} dx ,$$

kde $i = \sqrt{-1}$ je komplexní jednotka. Na obrázku 2.3-4 můžeme vidět funkci v časové oblasti (vlevo) a její Fourierův obraz.

Zpětná Fourierova transformace, přechod od frekvenční oblasti k prostorové, je definována vztahem

$$f(x) = \int_{-\infty}^{+\infty} F(u) e^{+i 2 \pi u x} du$$



Obr 2.3-4: Funkce $f(x)$ (vlevo) a její amplitudové spektrum $|F(u)|$

Fourierova transformace diskrétní funkce $I(x)$ se nazývá (*přímá a zpětná*) *diskrétní Fourierova transformace* (DFT) a má tvar

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} I(x) e^{-i 2 \pi \frac{ux}{N}} ,$$

$$I(x) = \sum_{u=0}^{N-1} F(u) e^{+i 2 \pi \frac{ux}{N}}$$

Fourierova transformace je dekompozicí původní funkce $I(x)$ na složení různě fázově posunutých sinusových funkcí s různou amplitudou. Máme-li bod o souřadnici u komplexního Fourierova obrazu, pak je tvořen reálnou částí, označme ji $Re(u)I$, a imaginární částí $Im(u)$. *Amplitudové spektrum* funkce $F(u)$ označme $|F(u)|$ a

$$|F(u)| = \sqrt{Re^2(u) + Im^2(u)}$$

a *fázové spektrum* je funkce

$$\varphi(u) = a \tan\left(\frac{Re(u)}{Im(u)}\right)$$

Fázové spektrum má vliv na fázové posuny jednotlivých sinusových složek. Amplitudové spektrum určuje amplitudy jednotlivých sinusových složek, a tedy tím i jejich vliv na celkový obraz.

3. Optický tok

Pro zjištění optického toku dvou po sobě následujících snímků, musíme nejprve definovat a zjistit pohyb těles. Při snímání scény pomocí videokamery se můžeme setkat se třemi druhy pohybu na scéně. První případ nastane, pokud máme kameru pevně ukotvenu a pohyb se děje na scéně. Nebo-li se pohybují tělesa vůči snímacímu zařízení. V druhém případě může být původcem pohybu samo snímací zařízení. Zařízení sice snímá statickou scénu, ale v důsledku jeho pohybu se objekty, potažmo celá scéna jeví v pohybu. Třetí a zároveň poslední případ nastává, když se zmíněné jevy projeví současně.

Pro ukázkou si můžeme vzít dva obrázky, zachycující pohybující se auto. Zjištěním, které pixely v obrázku se pohybují a které ne, dokážeme určit, jakou část obrazu tvoří nehybné pozadí a jaké pixely patří autu. Detailním studiem pohybu na obrázcích jsme schopni zodpovědět řadu otázek:

- Kolik pohybujících se předmětů scéna obsahuje.
- V jakém směru se dané tělesa pohybují.
- Zda se tělesa pohybují po lineární trajektorii, či rotují kolem své osy.
- Jak rychle tělesa mění svoji polohu.

Na základě těchto informací získaných ze sekvence snímků, definujeme *Optický tok obrazu*. Matematicky je optický tok vyjádřen polem vektorů. Každý vektor popisuje rychlost a směr, s jakou se jednotlivé oblasti v obraze pohybují vzhledem k snímacímu zařízení. Hodnoty těchto vektorů jsou závislé na době, po jakou je optický tok měřen.

Při hledání informací o tělesech v obraze se můžeme setkat s názvem obrazový tok. Jedná se také o pole dvou-rozměrných vektorů. Informace obsažené ve vektorech vyjadřují pohyb jednotlivých těles v obraze vůči obrazu a je to vlastně jeden z případů optického toku. Jeho hodnoty se však nemusí vždy shodovat s hodnotami optického toku. Matematický rozdíl mezi optickým a obrazovým tokem je rozdílný pouze pro pohled na větší plochy, ale ve skutečnosti jsou hodnotově velmi podobné. Ještě je důležité si připomenout, že optický tok nese časově závislé informace.

Ukažme si, jakou podobu má vektor optického toku v obrázku. Jedná se o třírozměrný vektor $I(x,y,t)$, kde x a y popisují souřadnice v obrázku a t vyjadřuje čas. Pro jednorozměrný obrázek byl měl vektor podobu $I(x,t)$.

Definujme si nyní optický tok pro obrazový bod (x,y) . Jde o dvourozměrný vektor určující rychlost

$$\mathbf{v} = \vec{i} v_x + \vec{j} v_y$$

kde \vec{i} a \vec{j} jsou jednotkové vektory, udávající pohyb v ose x a y pro blízké okolí bodu (x,y) a pro malou hodnotu t . Můžeme tedy psát, že

$$I(p+v_x\Delta t, q+v_y\Delta t, t+\Delta t)=I(p,q,t)$$

pro všechny (p,q) v blízkém okolí. Pro definici můžeme také použít pole vektorů

$$v = \vec{i}v_x(x,y,t) + \vec{j}v_y(x,y,t),$$

kde v_x a v_y jsou funkcemi x , y a t . Uvažujme ale jeden speciální případ nazývaný uniformní optický tok. V tomto případě jsou v_x a v_y konstantní pro x , y a t . Uniformní optický tok nastává v případě, kdy je nezměněný obrázek $I(x,y)$ posunut ve směru $v = \vec{i}v_x + \vec{j}v_y$. Uniformní optický tok je zajímavý, neboť zjednodušuje počítání optického toku na počítačích. Při výpočtech totiž předpokládáme, že v malých oblastech je optický tok neměnný.

3.1 Metody výpočtu optického toku

V poslední době prodělala videotechnika velký rozvoj a spolu s ní i obory, které nějakým způsobem pracují s obrazem. Při hledání metod pracujících s obrazem, můžeme narazit na značnou spoustu algoritmů, které obraz zpracovávají a upravují. Většina z nich však vychází ze dvou základních metod. První z nich se nazývá *Korelační metoda* a druhá se jmenuje *Gradientní metoda*.

3.1.1 Korelační metoda

Korelační metoda je možná nejpřímější metodou, jak získat optický tok ze souboru sekvenčních obrázků. Metoda je založena na detekci a porovnávání zajímavých míst v obraze. Nejprve vybere na prvním snímku malou oblast zajímavých pixelů. Tuto malou oblast posléze hledá v následujícím snímku. Rozdíl mezi pozicemi malých výřezů obrazu v prvním a druhém snímku nám definuje vektor vyjadřující optický tok pro danou oblast. Předpokladem pro tuto metodu je fakt, že pozorovaná část změní svůj povrch a tvar mezi dvěma pozorovanými snímky pouze velmi málo. Formálně bychom mohli zapsat výpočet optického toku (v_x, v_y) pro bod (p,q) takto. Vybereme zajímavé místo v obraze (p,q) , prohledáme jeho blízké okolí $N(p,q)$ a předpokládáme, že (v_x, v_y) patří do okolí $N(p,q)$ tohoto bodu. Hodnoty v_x a v_y nám vyjadřují posun v osách x a y v čase t . Pro získání optického toku celého obrazu prohledáváme všechna okolí zajímavých bodů a hledáme jejich obrazy v následujícím snímku. Potom optický tok vyjádříme jako

$$(v_x, v_y) = \min_{(v_x, v_y) \in N(p,q)} \sum_{i=-w}^w \sum_{j=-h}^h g(I(p-v_x+i, q-v_y+j, t-1), I(p+i, q+j, t))$$

kde $g(.,.)$ je funkce počítající vzdálenost a w a h určuje šířku a výšku prohledávaného okna. Pokud se dvě okna porovnaná korelací rovnají, pak

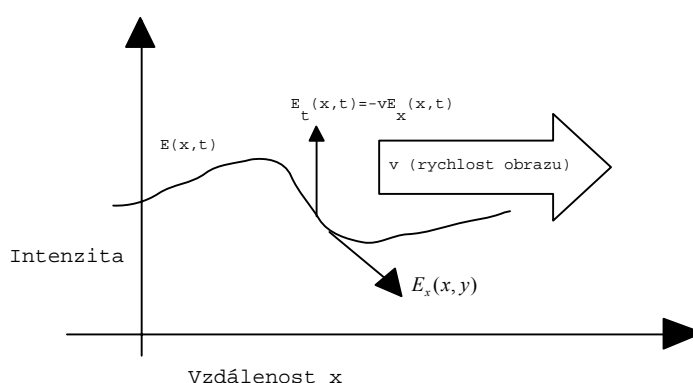
$$g(a,b) = -ab.$$

Pokud se dvě okna rovnají na základě Euklidově vzdálenosti, pak

$$g(a,b) = (a-b)^2$$

Abychom co možná nejvíce zamezili velkým chybám, snažíme se pracovat s malými oblastmi $N(p,q)$, ve kterých vzorek vyhledáváme a minimalizujeme časový rozdíl mezi sousedními snímky. Korelační metoda se nejvíce používá na vyhledávání optického toku v obrázcích znázorňující mraky nebo podobné objekty s nepříliš ostrými hranami.

3.1.2 Gradientní metoda



Obr.3.1.2-1 Gradientní metoda pro jednorozměrný obraz

Jak již název metody napovídá, pracuje se s gradientem. V tomto případě s gradientem jasu obrazu. Na obrázku 3.1.1-1 je zobrazena změna intenzity v čase t pro jednorozměrný obrázek. Vyjádřeme si jas obrazu v bodě (x,y) a čase t jako $E(x,y,t)$. Nyní si ukažme co se stane, když se vzor pohybuje. Vzor mění svoji pozici, ale jeho jas zůstává konstantní, nebo-li

$$\frac{dE}{dt} = 0.$$

Předpokládejme, že část obrazu se posunula o vzdálenost δx ve směru x a δy ve směru y za čas δt . Víme-li, že intenzita obrazu se v čase nemění, pak platí

$$E(x, y, t) = E(x + \delta x, y + \delta y, t + \delta t),$$

což je názorně vidět na obrázku 3.1.1-2. Rozepsáním pravé strany dostáváme rovnice

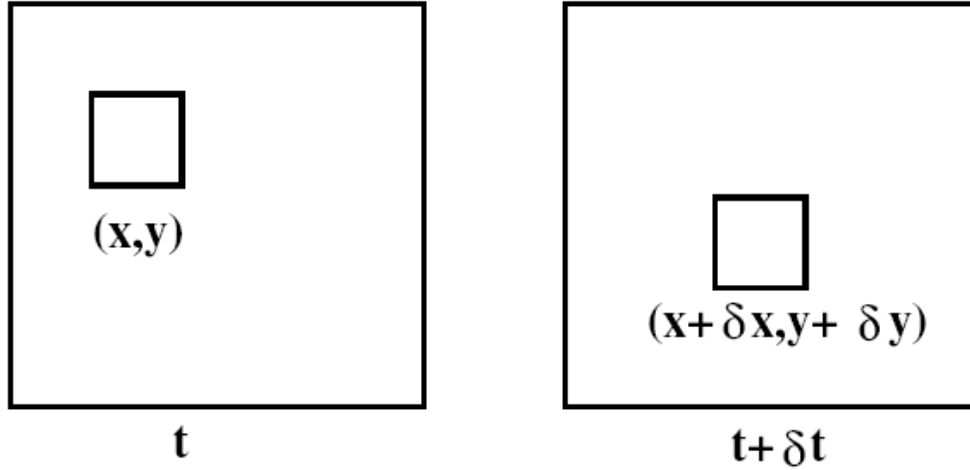
$$E(x, y, t) = E(x, y, t) + \delta x \frac{\partial E}{\partial x} + \delta y \frac{\partial E}{\partial y} + \delta t \frac{\partial E}{\partial t} + \varepsilon,$$

kde ε vyjadřuje hodnotu druhou a vyšší derivaci ∂x , ∂y a ∂t . Další úpravou vzorce dostáváme rovnici

$$\frac{\delta x}{\delta t} \frac{\partial E}{\partial x} + \frac{\delta y}{\delta t} \frac{\partial E}{\partial y} + \frac{\partial E}{\partial t} + O(\delta t) = 0,$$

kde $O(\delta t)$ je závislé na δt , a předpokládáme že δx , δy se mění s δt . Budeme-li čas minimalizovat, nebo-li $\delta t \rightarrow 0$, dostáváme

$$\frac{\partial E}{\partial x} \frac{dx}{dt} + \frac{\partial E}{\partial y} \frac{dy}{dt} + \frac{\partial E}{\partial t} = 0.$$



Obr. 3.1.2-2 Obrázek o souřadnicích (x, y, t) je stejný jako obrázek se souřadnicemi $(x + \delta x, y + \delta y, t + \delta t)$

Označme si nyní

$$u = \frac{dx}{dt} \quad \text{a} \quad v = \frac{dy}{dt}.$$

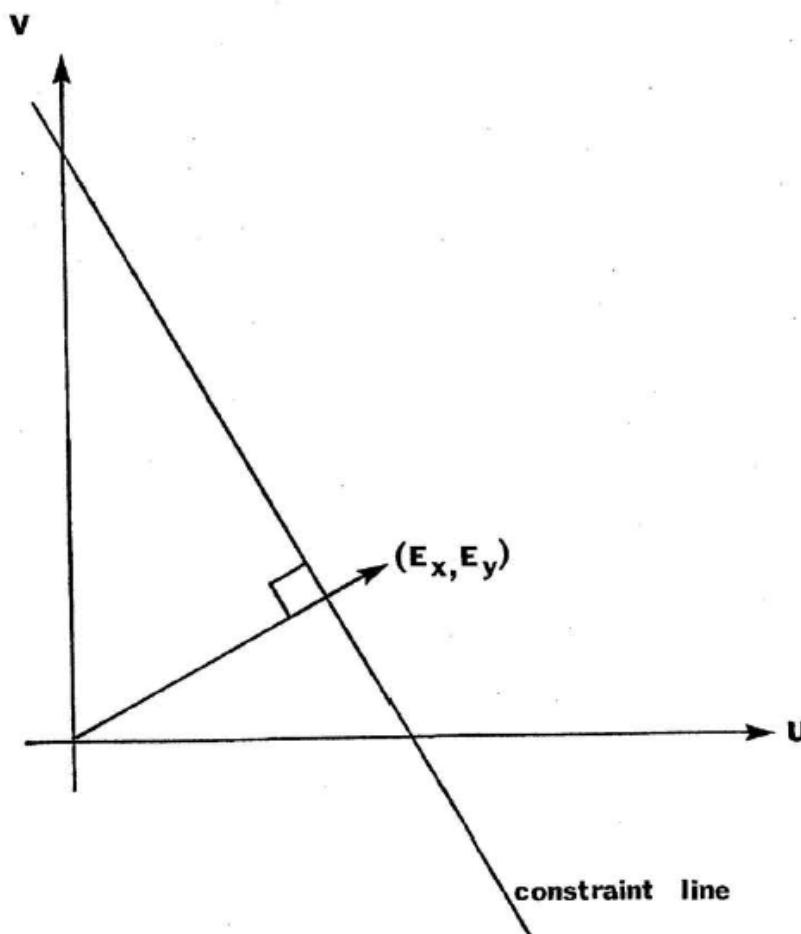
Při dosazení rovnic do předchozí rovnice, se vše na první pohled zjednoduší, ale zároveň zjišťujeme, že dostáváme jednu rovnici pro dvě neznámé u a v ,

$$E_x u + E_y v + E_t = 0,$$

jejíž řešení není nijak triviální.

Na obrázku 3.1.1-3 máme zobrazenou rychlost toku pro malou oblast. Tuto skutečnost můžeme také popsat rovnicí

$$(E_x, E_y) \cdot (u, v) = -E_t.$$



Obr. 3.1.2-3 znázorňuje rychlost změny jasu (E_x, E_y) v obraze. Tato změna určuje rychlost a směr optického toku.

Jak je z obrázku 3.1.1-3 vidět, určuje změna intenzity jasu v obraze velikost a směr optického toku sledované oblasti. Optický tok pro danou oblast můžeme vyjádřit rovnicí

$$-\frac{E_t}{\sqrt{E_x^2 + E_y^2}}.$$

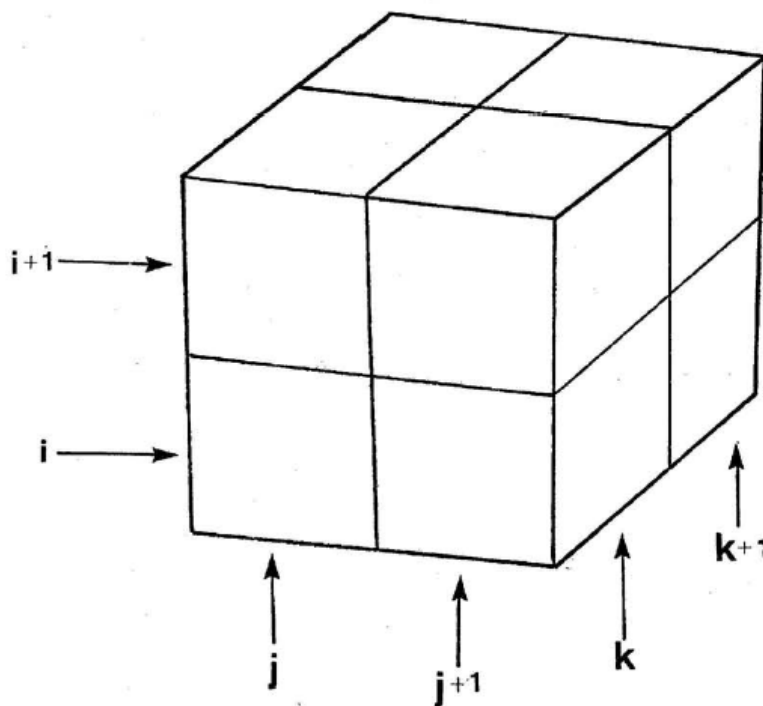
Při počítání optického toku nesmíme zapomínat na jednu důležitou věc. Optický tok nelze počítat pro jednotlivé části obrazu samostatně, ale musíme jej počítat jako celek a brát v úvahu, že jednotlivé části obrazu na sebe navzájem působí a ovlivňují se.

Abychom při výpočtu optického toku dosahovali nejlepších výsledků, je třeba, aby zpracovávaná data E_x , E_y , a E_t byla konzistentní. Uvedu-li to na příkladu, je třeba aby sekvence po sobě jdoucích obrázků zachycovala stejnou scénu, a aby bod x, y, t z prvního snímku korespondoval s bodem $x, y, t + \delta t$ v druhém snímku. Proč je pro nás tento předpoklad velmi důležitý si ukážeme na obrázku 3.1.1-4. Tento obrázek zachycuje časové a prostorové vztahy mezi hodnotami E_x , E_y a E_t . Výsledná intenzita jednotlivých bodů obrazu je totiž tvořena rozdílem okolních intenzit. Nebo-li

$$E_x \approx \frac{1}{4} \{ E_{i,j+1,k} - E_{i,j,k} + E_{i+1,j+1,k} - E_{i+1,j,k} + E_{i,j+1,k+1} - E_{i,j,k+1} + E_{i+1,j+1,k+1} - E_{i+1,j,k+1} \}$$

$$E_y \approx \frac{1}{4} \{ E_{i+1,j,k} - E_{i,j,k} + E_{i+1,j+1,k} - E_{i,j+1,k} + E_{i+1,j,k+1} - E_{i,j,k+1} + E_{i+1,j+1,k+1} - E_{i,j+1,k+1} \}$$

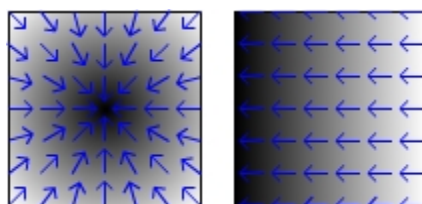
$$E_y \approx \frac{1}{4} \{ E_{i,j,k+1} - E_{i,j,k} + E_{i+1,j,k+1} - E_{i+1,j,k} + E_{i,j+1,k+1} - E_{i,j+1,k} + E_{i+1,j+1,k+1} - E_{i+1,j+1,k} \}$$



Obr. 3.1.2-4 – Prostorové vyjádření sekvence obrázků. Proměnná i značí x-ovou souřadnici obrazu, j značí y-ovou hodnotu obrazu a k znázorňuje časovou posloupnost snímků.

3.2 Výpočet gradientu

Jak ale získat místa kde je gradient a co to vlastně onen gradient je? V první řadě je třeba si uvědomit, že gradient souvisí s intenzitou vyzařováním, nebo odraženého světla. Tudíž pro jeho výpočet nám stačí pracovat s obrázky převedenými do stupňů šedi. Tato skutečnost nám umožňuje vypustit veškeré barevné informace z obrazu a tak i třeba snížit paměťové nároky výpočetních algoritmů. Matematicky je gradient popsán jako funkce, která udává změnu intenzity jasu v ose x a ose y . V každém bodě obrazu tak můžeme zjistit, jakým směrem a jak rychle se mění intenzita záření. Na obrázku 3.2-1 můžeme vidět změnu intenzity světla.



Obr. 3.2-1 – Změna intenzity světla v obraze.

Pro výpočet gradientu nám slouží řada operátorů. Jedním z nich je i *Sobelův operátor*, kterého jsem využil i ve svých výpočtech. Jedná se o dvě matice 3x3. Na obrázku 3.2-2 jsou zobrazeny matice použité pro výpočet horizontálního a druhá pro výpočet vertikálního gradientu.

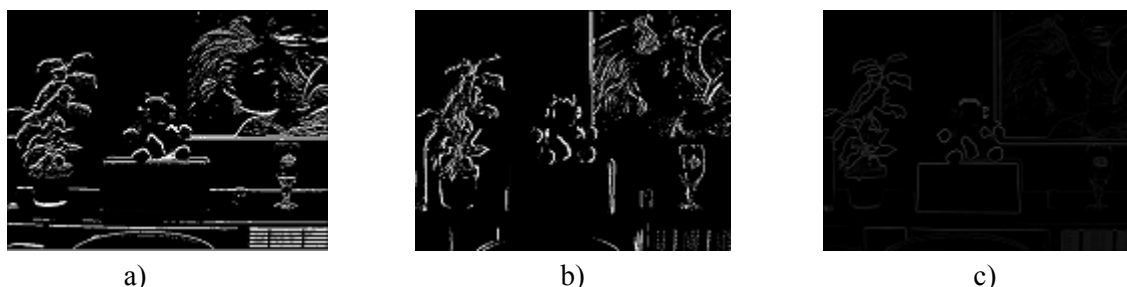
$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \text{a} \quad G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Obr. 3.2-2 Matice Sobelova operátoru pro výpočet gradientu ve směru osy x a y

Tyto matice se aplikují na zdrojový snímek a jejich výsledkem je zvýraznění míst, kde dochází k ostrému přechodu mezi světlými a tmavými místy či naopak (Obr. 3.2-1). Zároveň jsou potlačeny informace z míst, kde je změna intenzity pouze nepatrná. Proces aplikace operátoru na výpočet gradientu se nazývá *konvoluce*. Pokud bychom chtěli získat gradient intenzity záření celého obrazu a ne jenom jeho x-ových či y-ových složek, můžeme výsledky z os x a y sjednotit dohromady aplikací rovnice

$$G = \sqrt{G_x^2 + G_y^2}$$

kde G je výsledná intenzita v jednotlivých bodech obrazu a G_x a G_y jsou intenzity jasů v jednotlivých osách. Dojde tak ke zvýraznění místa, kde mají obě složky svůj podíl. Na obrázku 3.2-3 je vidět aplikování matic pro zjištění gradientu v osách x a y a výpočet celkového gradientu obrazu.



Obr. 3.2-3 – Aplikace Sobelova operátoru, a) ve směru osy x, b) ve směru osy y, c) výsledné zobrazení gradientu celého obrazu.

3.3 Horn a Schunck algoritmus

Na světě již bylo publikováno řada algoritmů, které se zabývají zjišťováním a výpočtem optického toku obrazu. Jedním z velmi známých a úspěšných algoritmů řešících tento problém, prezentovali pánové Horn a Schunck v roce 1980. K výpočtu optického toku obrazu se vychází z již dříve zmíněné *Gradientní metodu*. Jelikož je výpočet založen na této metodě, platí pro něj stejná omezení, jaká platily pro gradientní metodu. Při nedodržení podmínek, nastává chybná klasifikace optického toku a tudíž algoritmus nevrací použitelné hodnoty. Hlavní myšlenkou algoritmu, je

postupné zpřesňování optického toku jednotlivých bodů obrazu na základě hodnot blízkého okolí zkoumaného bodu.

Samotný výpočet Horn a Schunck algoritmem, probíhá v několika krocích. Nejprve je třeba určit hodnotu E_x , E_y a E_t , vyjadřující intenzitu v bodě x , y a čase t . Tento postup je popsán v kapitole 3.1.2 gradientní metoda a je zobrazen na obrázku 3.1.2-4. V této chvíli známe pouze změnu intenzit jednotlivých bodů v osách x a y . Optický tok bodů obrazu je prozatím nulový. V dalším kroku, již dochází k samotnému výpočtu *optického toku* jednotlivých bodů obrazu. Vypočítáme si hodnoty vektorů \bar{u} a \bar{v} , jakožto průměrnou hodnotu vektorů u a v čtyř okolních bodů. Tento krok je potřebný z jednoho prostého důvodu. Jak jsem již dříve zmiňoval, gradientní metoda nepočítá optický tok jako hodnotu jednoho samostatného místa v obraze, ale jako hodnotu, jež je ovlivněna svým okolím. Hodnotu vektoru \bar{u} a \bar{v} tedy dostaneme, provedeme-li konvoluci bodu o souřadnicích x , y s maticí o rozměrech 3x3

$$\bar{u} = \frac{1}{4} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} * u(x, y)$$

a

$$\bar{v} = \frac{1}{4} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} * v(x, y)$$

Nové hodnoty u a v jsou získány výpočtem rovnic

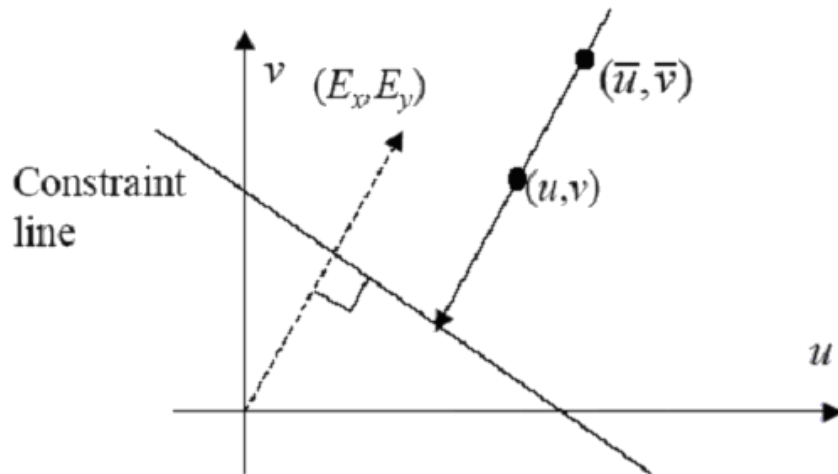
$$u_{ij}^{n+1} = \bar{u}_{ij}^n - \alpha E_x$$

$$v_{ij}^{n+1} = \bar{v}_{ij}^n - \alpha E_y,$$

kde n označují stupeň iterace, takže u_{ij}^0 a v_{ij}^0 vyjadřují počáteční hodnotu optického toku a \bar{u} a \bar{v} značí průměrnou hodnotu okolních u a v . Hodnotu α získáme výpočtem rovnice

$$\alpha = \frac{E_x \bar{u}_{ij}^n + E_y \bar{v}_{ij}^n + E_t}{1 + \lambda(E_x^2 + E_y^2)}$$

kde λ ovlivňuje hladkost optického toku. Na obrázku 3.3-1 je vidět, jak je nová hodnota (u, v) v daném bodě, rovna průměru okolních hodnot, korigovaná směrem gradientu jasu v daném místě.



Obr3.3-1 Výpočet nové hodnoty (u, v) na základě průměrných hodnot okolí a směru gradientu jasu v daném místě.

Výpočet hodnot \bar{u} a \bar{v} a u a v provádíme v několika opakováních dokola, abychom jejich velikost postupně zpřesňovali.

Formálně by se dal popsáný algoritmus zapsat takto:

begin

for $i=1$ to N do

for $j=1$ to M do begin

vypočítáme si hodnoty $E_x(i, j, t)$, $E_y(i, j, t)$ a $E_t(i, j, t)$;

nastavíme hodnoty $u(i, j)$ a $v(i, j)$ na nulu;

end; {for}

nastavíme hodnotu λ ;

nastavíme počet iteračních kroků n_0 ;

aktuální iterace $n=1$;

while $n \leq n_0$ do begin

for $i=1$ to N do

for $j=1$ to M do begin

$$\bar{u} = \frac{1}{4}(u(i-1, j) + u(i+1, j) + u(i, j-1) + u(i, j+1)) ;$$

$$\bar{v} = \frac{1}{4}(v(i-1, j) + v(i+1, j) + v(i, j-1) + v(i, j+1));$$

$$\alpha = \frac{E_x(i, j, t)\bar{u} + E_y(i, j, t)\bar{v} + E_t(i, j, t)}{1 + \lambda(E_x^2 + E_y^2)} ;$$

$$u(i, j) = \bar{u} - \alpha E_x(i, j, t) ;$$

$$v(i, j) = \bar{v} - \alpha E_y(i, j, t) ;$$

```
end; {for}  
n = n+1;  
end; {while}  
end;
```

Výsledek, který získáme lze ovlivnit několika faktory. Podle počtu zvolených iteračních kroků, jsme schopni minimalizovat chybu. Pokud kroků bude příliš málo, chyba bude velká, pokud bychom nechali běžet proces minimalizace chyby zbytečně dlouho, nedosáhli bychom již lepšího výsledku, ale pouze by narůstal čas programu bez již větších účinků na výsledek. Druhým prvkem, kterým můžeme ovlivnit výsledek je parametr λ . Tato hodnota určuje jakou hladkost optického toku.

4. Závěr

4.1 Zhodnocení vlastní práce

Při testování algoritmu Horn a Schunck jsem zjistil, že tato metoda je vhodná na obrázky zachycující jasně ohraničené předměty. Při aplikaci na snímky typu marky nebo na scénu s nepříliš jasnými objekty, docházelo k nepřesnému selhávání a přílišným chybám při určování o optickém toku jednotlivých částí obrazu. To mělo za následek, že i celkový optický tok obrazu nebylo možno dobře určit. Dále jsem při testování zkoušel měnit hodnoty parametru λ a $n\theta$. Počet iteračních kroků se jevil neoptimálnější okolo hodnoty 100. Při několikanásobně menší hodnotě byl vypočítaný optický tok nepříliš přesný a při dalším zvětšování se již hodnoty nezpřesňovaly. Parametr λ jsem měnil v rozmezí 1-10, ale jeho vliv na celkový výsledek je mnohem menší, než vliv počtu iteračních kroků.

4.2 Možnosti dalšího rozvoje

V současné chvíli projekt pracuje pouze se dvěma obrázky o rozměrech 640x480 pixelů. Jako další možné rozšíření vidím oproštění se od pevně zadané velikosti vstupních dat a také práce se sekvencí obrázků, nikoli pouze se dvěma.

Literatura

- [1] J.Žára, B. Beneš, P.Felkel. Moderní počítačová grafika, Computer Press, 1998
- [2] pdf soubor Determining Optic Flow, Berthold K.P. Horn and Brian G. Schunck
people.csail.mit.edu/bkph/papers/Optical_Flow_OPT.pdf
- [3] WWW stránky. <http://homepages.inf.ed.ac.uk/rbf/HIPR2/sobel.htm> (duben 2006)
- [4] WWW stránky <http://en.wikipedia.org/wiki/Gradient> (duben 2006)
- [5] WWW stránky <http://www.mathworks.com/access/helpdesk/help/toolbox/vipblks/opticalflow.html>
- [6] pdf soubor Tutorial: Computing 2D and 3D Optical Flow, J.L.Barron and N.A.Thacker. 2005
www.tina-vision.net/docs/memos/2004-012.pdf