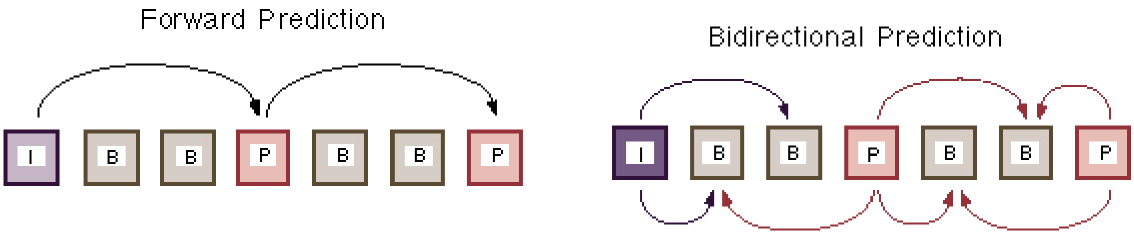
8.2. Časová interpolácia, odhad pohybu

Časová, resp. medzisnímková interpolácia je interpolácia sekvencie snímok. Snímky idúce za sebou v sekvencii sú si navzájom veľmi podobné. Práve táto vlastnosť, teda vysoká korelácia medzi snímkami v sekvencii videa umožňuje efektívne kódovanie znižovaním časovej redundancie. Metódy kódovania videa s kompenzáciou pohybu sa často využívajú na skúmanie časovej redundancie medzi nasledujúcimi snímkami.

Medzisnímková interpolácia spočíva vo vytváraní nových snímok a tým „hustejšieho toku snímok videa.

Nové snímky obrazu môžeme vytvoriť z už existujúcich snímok v sekvencii. Väčšinou používame dve susedné snímky - jednu predchádzajúcu a jednu nasledujúcu.

V štandarde MPEG sa využíva medzisnímková interpolácia na kódovanie sekvencie. Z hľadiska kódovania rozlišujeme 3 typy smímok: **I,P,B**



Obr. 8.4. Typy snímok v štandarde MPEG

I – (Intraframe) snímka, kde bola použitá len priestorová interpolácia (žiadny odhad pohybu ani rozdielové hodnoty)

P – (forward Prediction) - rozdielová snímka získaná predikciou z predchádzajúcej snímky

B (Bidirection prediction) – snímka získaná z jednej I- snímky a jednej P-snímky

Štandardná štruktúra MPEG stream "Group of Pictures" (GOP) IBBPBBPBBPBBPBB

PRIDRŽANIE NULTÉHO RÁDU (ZERO-ORDER HOLD)

Je to najjednoduchšia a v praxi často používaná metóda – novú snímku vytvárame opakovaním existujúcej snímky, ktorá je najbližšie v čase.

Umožňuje napríklad transformáciu dynamického obrazu z 24 snímok/sek. na

signál 60 polsnímok (fields) /sek. a to tak, že sú vytvorené

- tri po sebe idúce polia (polsnímky) z jednej snímky pohyblivého obrazu a
- ďalšie dve po sebe idúce polia z nasledujúcej snímky

Tento proces sa opakuje pre celú sekvenciu snímok – jeho originálny názov je

"3:2 pull-down" metóda.

Výsledky sú použiteľné hlavne pri scénach, v ktorých sa nevyskytuje veľký globálny pohyb.

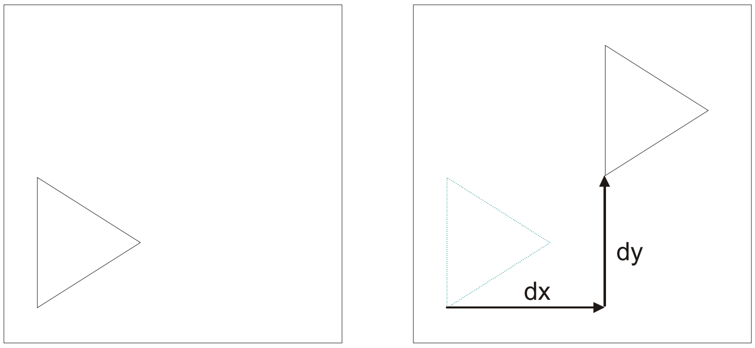
Pri veľkom globálnom pohybe v obraze však môže "zero-order hold" metóda spôsobiť „trhanie“ obrazu.

Na zvýšenie účinnosti medzisnímkovej interpolácie môžeme použiť kompenzáciu pohybu.

Odhad pohybu je proces, pri ktorom odhadujeme zmenu polohy objektu v postupnosti snímok.

Väčšina zmien v obraze medzi dvomi po sebe idúcimi snímkami dynamického obrazu je zapríčinená pohybom objektu v obraze.
Spracovanie obrazu podmienené prítomnosťou pohybu v obraze nazývame **spracovanie obrazu s kompenzáciou pohybu.**

Odhad pohybu pre **POSUVNÝ POHYB OBJEKTU**



(a) (b)

Obr. 8.5 posuvný pohyb objektu

Posunutie obrazu v smere vektora (d_x, d_y) *medzi snímkami (a) a (b):*

a) $f(x, y, t_{-1})$, b) $f(x, y, t_0)$.

$f(x, y, t_{-1})$ – intenzita obrazu v čase t_{-1}, t.j. predošlá snímka

$f(x, y, t_0)$ – intenzita obrazu v čase t_0, t.j. aktuálna snímka

d_x – parameter vyjadrujúci horizontálny posuv medzi t_{-1} a t_0

d_y – parameter vyjadrujúci vertikálny posuv medzi t_{-1} a t_0

$$f(x, y, t_0) = f(x - d_x, y - d_y, t_{-1}) \qquad \text{(+)}$$

Ak predpokladáme rovnomerný pohyb, potom

$$f(x, y, t) = f(x - \mathrm{v}_\mathrm{x}(t - t_{-1}), y - \mathrm{v}_\mathrm{y}(t - t_{-1}), t_{-1}), \qquad t_{-1} \leq t \leq t_0$$

kde v_x je konštantná horizontálna rýchlosť pohybu

v_y je konštantná vertikálna rýchlosť pohybu

Vzťah v_x a v_y k parciálnym deriváciám:

$$\partial f(x, y, t)/\partial x, \ \partial f(x, y, t)/\partial y, \ \partial f(x, y, t)/\partial t$$

v priestorovo-časovej oblasti za predpokladadu rovnomerného pohybu popisuje diferenciálna rovnica.

Označme si $f(x,y,t_{-1})$ ako $s(x,y)$: $\qquad f(x,y,t_{-1}) = s(x,y)$

$$\alpha(x,y,t) = x - v_x(t - t_{-1}),\ \beta(x,y,t) = y - v_y(t - t_{-1})$$

Potom $\qquad f(x,y,t) = s(\alpha(x,y,t), \beta(x,y,t)) \qquad t_{-1} \le t \le t_0$

Za predpokladu, že existujú derivácie

$\partial f(x,y,t)/\partial x$, $\partial f(x,y,t)/\partial y$ a $\partial f(x,y,t)/\partial t$, dostaneme:

$$\frac{\partial f(x,y,t)}{\partial x} = \frac{\partial s}{\partial \alpha}\frac{\partial \alpha}{\partial x} + \frac{\partial s}{\partial \beta}\frac{\partial \beta}{\partial x} = \frac{\partial s}{\partial \alpha}$$

$$\frac{\partial f(x,y,t)}{\partial y} = \frac{\partial s}{\partial \alpha}\frac{\partial \alpha}{\partial y} + \frac{\partial s}{\partial \beta}\frac{\partial \beta}{\partial y} = \frac{\partial s}{\partial \beta}$$

$$\frac{\partial f(x,y,t)}{\partial t} = \frac{\partial s}{\partial \alpha}\frac{\partial \alpha}{\partial t} + \frac{\partial s}{\partial \beta}\frac{\partial \beta}{\partial t} = -v_x\frac{\partial s}{\partial \alpha} - v_y\frac{\partial s}{\partial \beta}$$

Po dosadení do poslednej rovnice platí **rovnica priestorovo-časového ohraničenia:**

$$v_x\frac{\partial f(x,y,t)}{\partial x} + v_y\frac{\partial f(x,y,t)}{\partial y} + \frac{\partial f(x,y,t)}{\partial t} = 0$$

(++)

Metódy využívajúce priestorovo-časové ohraničenie (++)

Na rovnicu priestorovo-časového ohraničenia sa môžeme pozerať ako na

lineárnu rovnicu o dvoch neznámych v_x a v_y, za predpokladu, že sú definované parciálne derivácie

$$\partial f(x,y,t)/\partial x,\ \partial f(x,y,t)/\partial y \text{ a } \partial f(x,y,t)/\partial t.$$

Ak vypočítame jednotlivé derivácie v bodoch (x_i, y_i, t_i) pre $1 \le i \le N$,

v ktorých v_x a v_y považujeme za konštantné, získame množinu lineárnych rovníc:

$$v_x \left.\frac{\partial f(x,y,t)}{\partial x}\right|_{(x_i,y_i,t_i)} + v_y \left.\frac{\partial f(x,y,t)}{\partial y}\right|_{(x_i,y_i,t_i)} + \left.\frac{\partial f(x,y,t)}{\partial t}\right|_{(x_i,y_i,t_i)} \cong 0; 1 \le i \le N$$

Odhady rýchlosti získame minimalizovaním výrazu

$$Chyba = \sum_{i=1}^{N} \left[v_x \left.\frac{\partial f(x,y,t)}{\partial x}\right|_{(x_i,y_i,t_i)} + v_y \left.\frac{\partial f(x,y,t)}{\partial y}\right|_{(x_i,y_i,t_i)} + \left.\frac{\partial f(x,y,t)}{\partial t}\right|_{(x_i,y_i,t_i)} \right]^2$$

Predpoklady

jednoduchého posunu, ktorý viedol k **(+)** a

posunu konštantnou rýchlosťou, ktorý viedol k **(++)**, sú veľmi obmedzujúce.

- neumožňujú rotáciu objektu, zväčšovanie, alebo pohyb viacerých objektov rôznymi rýchlosťami v_x a v_y

Uvedené vzťahy platia

- pre oblasti pozadia, ktoré nie sú ovplyvnené pohybom objektov a
- pre oblasti obsadené objektami pohybujúcimi sa rovnomernou rýchlosťou pri predpoklade výhradne lokálneho rovnomerného pohybu a pri odhade dvoch parametrov pohybu (d_x, d_y) alebo (v_x, v_y) v každom bode alebo v každej malej časti obrazu.

Rozlišujeme 2 základé skupiny metód odhadu pohybu

- metódy odhadu pohybu podľa (+) – založené na porovnávaní oblastí
- metódy odhadu pohybu podľa (++) – vychádzajúce z priestorovo–časových obmedzení – snažíme sa odhadnúť rýchlosť pohybu v_x a v_y zo sústavy lin. rovníc (++)

METÓDY POROVNÁVANIA OBLASTÍ

– máme polohu objektu (vzor) na určitej snímke

– chceme určiť jeho polohu na predchádzajúcej alebo nasledujúcej snímke

Vektor posuvu (d_x, d_y) získame minimalizáciou chyby odhadu

$$Chyba = \iint_{(x,y)\in R} C\left[f(x,y,t_0), f(x-d_x, y-d_y, t_{-1})\right] dxdy$$

R – oblasť obrazu použitá na odhad (d_x, d_y) a

$C[\cdot,\cdot]$ – miera, ktorá vyjadruje vzdialenosť medzi dvomi argumentmi

Ak je $f(x,y,t)$ vzorkovaná v priestorových súradniciach (x,y) ➔ integrály môžu byť nahradené sumou.

Ak odhadujeme (d_x, d_y) v čase t_0 ➔ R je oblasť obrazu, ktorá obklopuje príslušnú priestorovú pozíciu, v ktorej robíme odhad vektora (d_x, d_y).

Veľkosť oblasti *R* je dôležitý parameter

ak je príliš veľká – predpoklad, že vektor (d_x, d_y) je približne konštantný v celej

oblasti R, nemusí byť správny a

– určenie chybového výrazu vyžaduje veľa výpočtov

ak je príliš malá – odhady môžu byť veľmi citlivé na šum

Rovnako existuje veľa rôznych alternatív miery $C[\cdot,\cdot]$.

Dve bežne používané miery $C[\cdot,\cdot]$:

1. kvadratická miera (druhá mocnina rozdielu).

$$Chyba = \iint_{(x,y)\in R} \left[f(x,y,t_0) - f(x-d_x, y-d_y, t_{-1})\right]^2 dxdy$$

2. absolútny rozdiel argumentov

$$Chyba = \iint_{(x,y)\in R} \left| f(x,y,t_0) - f(x-d_x, y-d_y, t_{-1})\right| dxdy$$

Výraz $f(x,y,t_0)-f(x-d_x,y-d_y,t_{-1})$ sa nazýva ***rozdiel posunutého rámca***.

Minimalizácia chyby odhadu predstavuje nelineárny problém.

Pokusy riešiť tento nelineárny problém viedol ku vzniku mnohých variácií odhadu pohybu:

- metódy porovnávania blokov a
- rekurzívne (iteračné) metódy

Obidve skupiny sa navzájom prelínajú.

METÓDY POROVNÁVANIA BLOKOV (ITU-T H.261, MPEG-1/2)

minimalizačný problém

Základné pravidlá

– stanovenie chyby pre každé (d_x, d_y) v rámci „rozumného“ intervalu a

–voľba takého vektora (d_x, d_y), pre ktorý je chyba minimálna

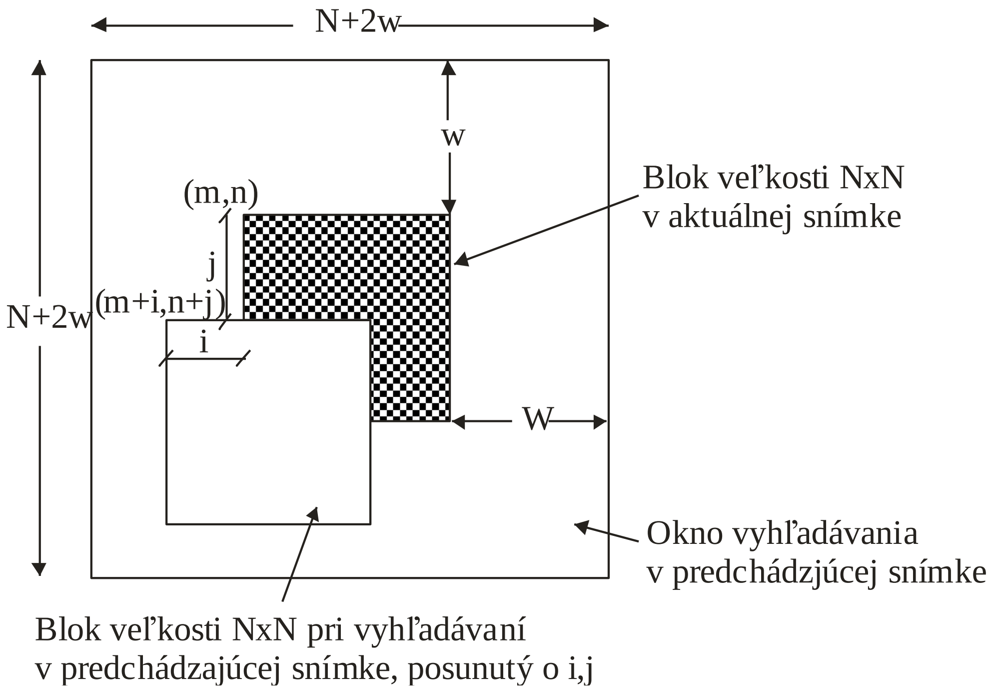
– blok intenzít bodov obrazu v čase t_0 priamo porovnávame s blokom v čase t_{-1}

Možné zjednodušenie:

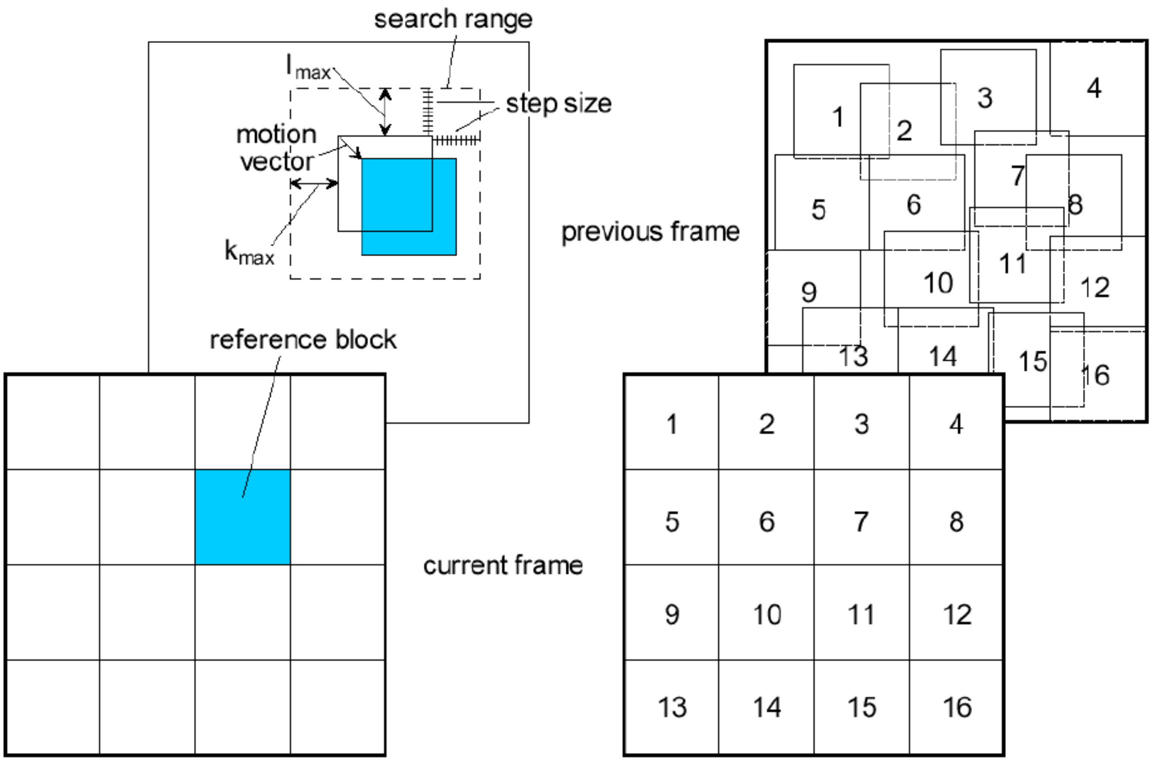
– vychádzame z predpokladu, že (d_x, d_y) je konštantné v rámci bloku veľkosti povedzme 7x7 bodov;to umožní redukciu množstva výpočtov

– limitovanie hodnôt prvkov vektora na celé hodnoty -ďalšia redukcia

– predikcia vektora pohybu podľa okolitých blokov



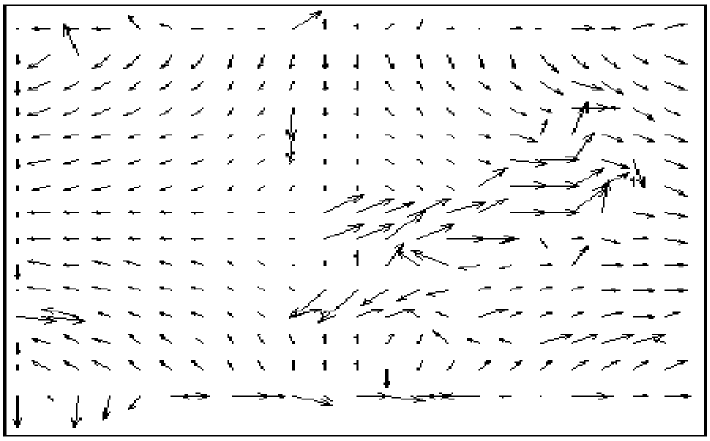
Obr. 8.6 Odhad pohybu – určenie zmeny polohy bloku obrazu medzi predchádzajúcou **Obr.**



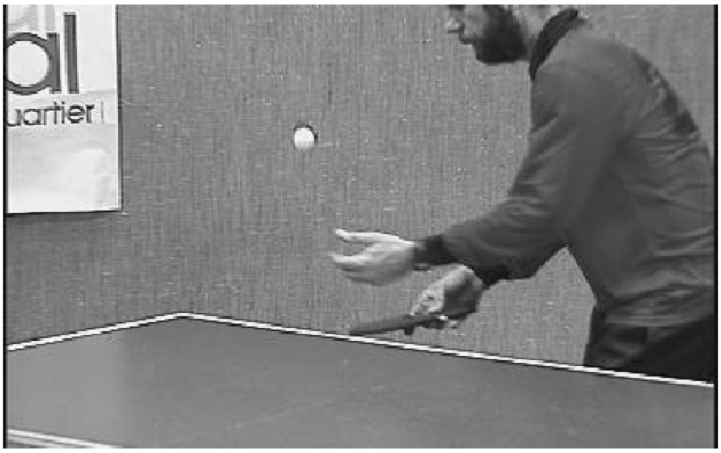
8.6 Odhad pohybu – určenie zmeny polohy blokov obrazu medzi predchádzajúcou

a aktuálnou snímkou a aktuálnou snímkou

Pozn.: Vzájomné prekrytie blokov pri prediktívnom kódovaní nevadí, pretože počítame polohu blokov pre aktuálnu snímku z predchádzajúcej, t.j. aktuálna snímka je v poriadku.



(a)



(b)

8.7 (a)Vektory pohybu pre snímku (b)

<u>Chybové funkcie na výpočet chyby odhadu:</u>

Stredná kvadratická odchýlka MSD (Mean Squared Difference)

$$MSE(m,n,k,l) = \frac{1}{N^2}\sum_{i,j=1}^{N}(f_{-1}(i+m, j+n) - f_0(i+k, j+l))^2$$

f_{-1} – predchádzajúca snímka a ***f_0*** je aktuálna snímka

k,l – pozícia makrobloku v aktuálnej snímke

m,n – pozícia referenčného makrobloku v predchádzajúcej snímke

NxN – veľkosť makrobloku

Absolútny rozdiel SAD (Sum of Absolute Differences):

$$SAD(m,n,k,l)=\sum_{i,j=1}^{N}\left|f_{-1}(i+m,j+n)-f_0(i+k,j+l)\right|$$

Miera závislosti 2 náhodných premenných:

korelácia CCF (Cross–Corelation Function), kovariancia – účinnejšie kritériá napr. v prípade prudkej zmeny osvetlenia (rozsvietenie/zhasnutie lampy)

Rýchle algoritmy odhadu pohybu

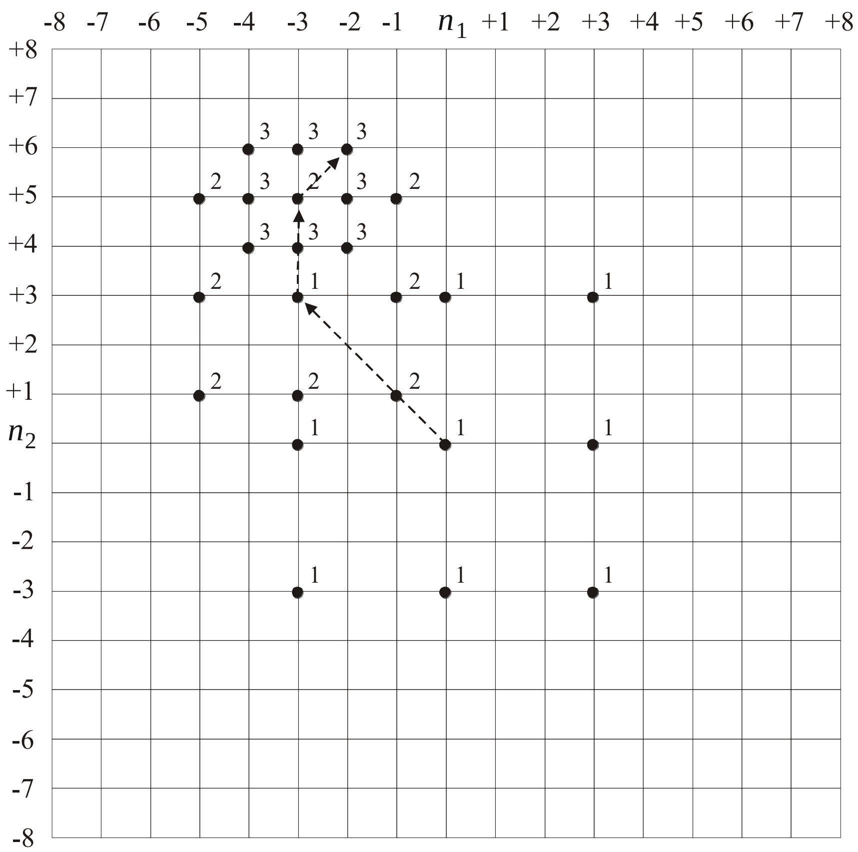
TROJKROKOVÁ VYHĽADÁVACIA METÓDA

– efektívna vyhľadávacia procedúra –

– umožňuje zníženie výpočtových nárokov pri porovnávaní blokov

Oblasť prehľadávania: p = 6

1. krok: chybu odhadu počítame pre 9 hodnôt posunu (d_x, d_y), (1). Medzi týmito deviatimi hodnotami si zvolíme (d_x, d_y) s najmenšou chybou. Predpokladajme, že najmenšia chyba je v bode $(d_x=-3, d_y=3)$.



Obr. 8.8 Trojkroková vyhľadávacia metóda – ilustračný obrázok

2. krok: vypočítame chybu odhadu v 8 bodoch, ktoré sú označené číslom (2), t.j. v polovičnej vzdialenosti od najlepšieho dosiahnutého výsledku. Opäť zvolíme (d_x, d_y) minimalizovaním deviatich chybových hodnôt – ôsmich nových hodnôt a chyby v bode $(d_x = -3, d_y = 3)$. Výsledok nech je $(\boldsymbol{d}_x = -3, \boldsymbol{d}_y = 5)$

3. krok: zopakujeme ešte jedenkrát. Na konci tretieho kroku máme odhad pohybu (d_x, d_y).

Oblasť prehľadávania: p = 6

Alternatíva: oddelený odhad d_x pomocou hľadania vektora $(d_x, 0)$.

Výhodou tejto metódy je, že odhadujeme len jeden parameter, čo je oveľa jednoduchšie, ako odhadovať naraz dva parametre.

REKURZÍVNE (ITERAČNÉ) METÓDY

Pri metóde porovnávania blokov hľadáme najlepšie $(\hat{d}_x, \hat{d}_y)$ pomocou explicitne stanovenej chyby pre špecifikovanú množinu (d_x, d_y).

Alternatíva – zostupné algoritmy – ***napríklad rekurzívne metódy***.

Nech $(\hat{d}_x(k), \hat{d}_y(k))$ označuje odhad (d_x, d_y) po k -tej iterácii.

V rekurzívnych metódach získame odhad (d_x, d_y) po $(k+1)$ -vej iterácii, $(\hat{d}_x(k+1), \hat{d}_y(k+1))$, zo vzťahov:

$$\hat{d}_x(k+1) = \hat{d}_x(k) + u_x(k)$$

$$\hat{d}_y(k+1) = \hat{d}_y(k) + u_y(k)$$

$u_x(k)$ a $u_y(k)$ – aktualizácia po iterácii

Určenie aktualizačných vzťahov závisí od použitej metódy.

Vyhladzovanie obrazu pred odhadom pohybu často zvyšuje výkon rekurzívnych metód

Hlavná nevýhoda metód porovnávania oblastí: **vyžadujú si veľké množstvo výpočtov**

Algoritmy prehľadávania (Block Matching Algorithms)

1. úplné prehľadávanie – optimálne z hľadiska presnosti odhadu pohybu
2. trojkrokový algoritmus –
3. 4 – krokový algoritmus –
4. 2D logaritmické prehľadávanie –

   suboptimálne z hľadiska presnosti odhadu pohybu
5. Binárne prehľadávanie –
6. OTA – smerové prehľadávanie –
7. Ortogonálne prehľadávanie –
8. Špirálové prehľadávanie –

Využitie odhadu pohybu:

- prediktívne medzisnímkové kódovanie – kompenzácia pohybu
- segmentácia videa
- medzisnímková interpolácia – vytvorenie chýbajúcich snímok
– oprava poškodených snímok

Vychádzame z predpokladu, že správny odhad pohybu je niekde v blízkosti približného odhadu pohybu – nemusí vždy platiť – napr. nájdeme lokálne minimum namiesto globálneho minima

Možná modifikácia: (tzv. Dynamic Search Window Algorithms) – berieme do úvahy 2 najlepšie odhady a nie len najlepší odhad

Pre odhad pohybu bloku sa dá využiť známy odhad pohybu okolitých blokov