Képfeldolgozás
7. hét

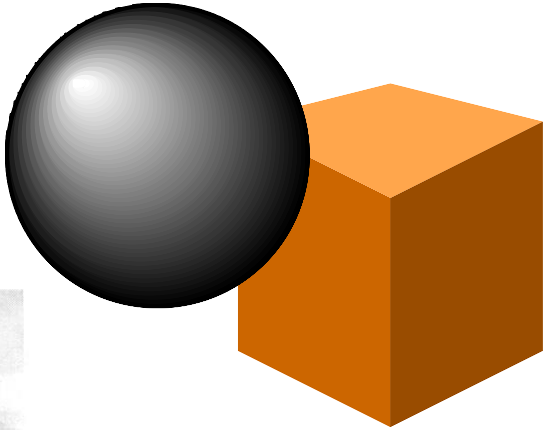
Mozgás felhasználása a képfeldolgozásban
Mozgásparallaxis
Mozgásérzékelés differenciaképek segítségével
Zajszűréssel egybekötött mozgásérzékelés
Optical flow
Mozgás alapú szegmentálás

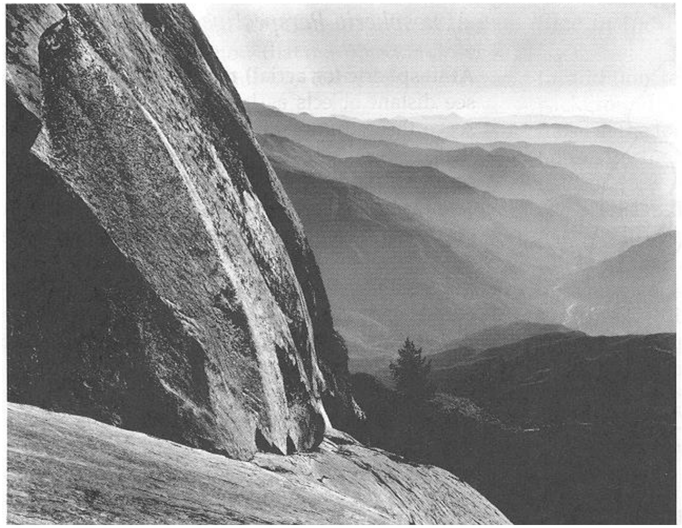
Az előző rész tartalmából…

Térlátás összetevői

Mai ismereteink szerint három kategóriába sorolhatók:

1. monokuláris („egy szemmel”)
   - Lineáris perspektíva
   - A tárgy elvárt mérete és textúra torzítás
   - Árnyalás és takarás
   - Légköri torzítások
2. extraretinális (nem látványból származó)
3. binokuláris („két szemmel”)

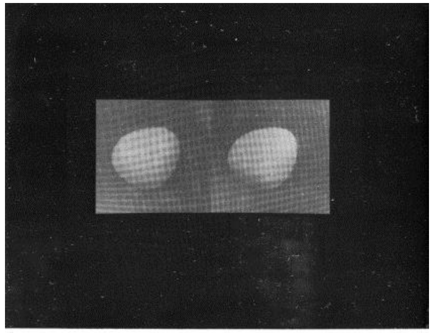


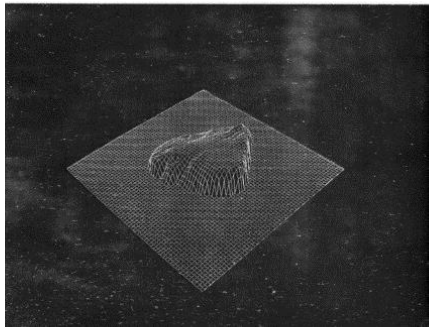


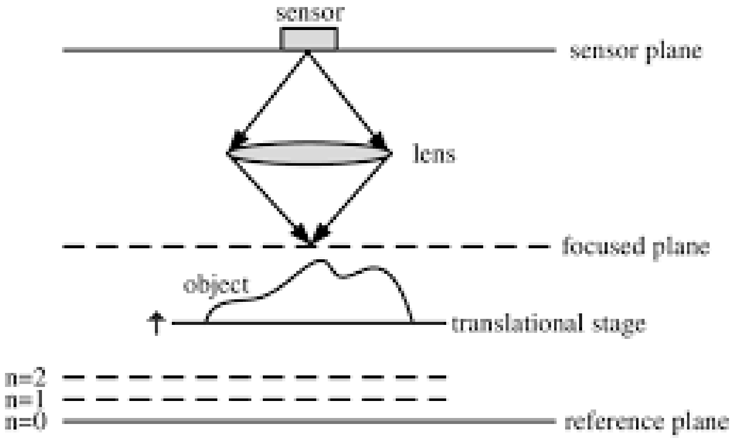
Műszaki megoldások

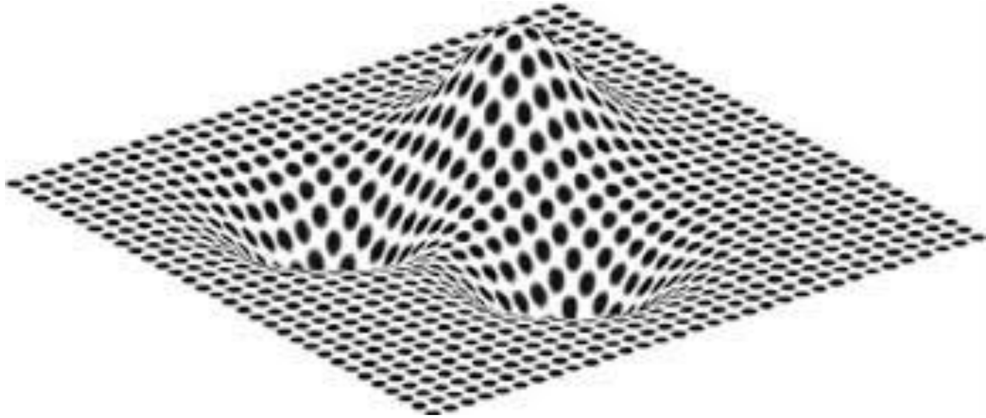
Shape from X, ahol az X:

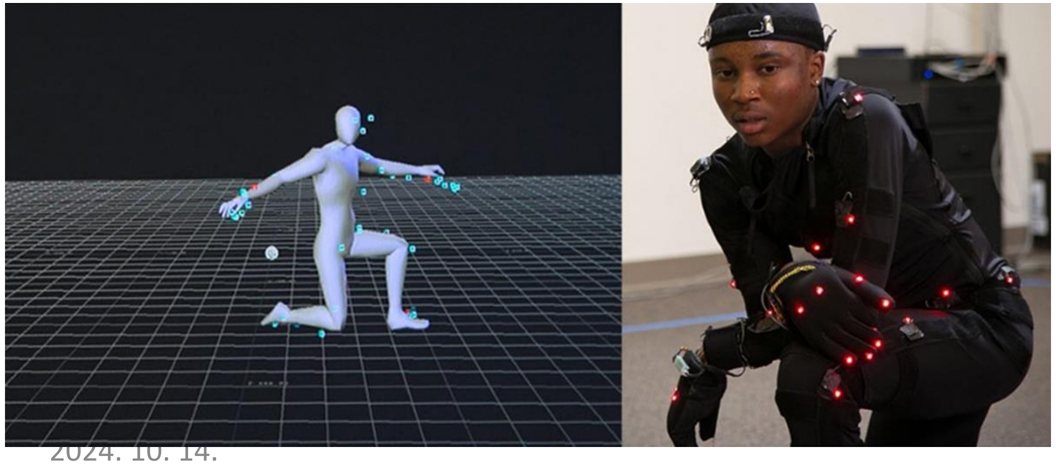
- Shading
- Focus
- Texture
- Motion
- Stereo

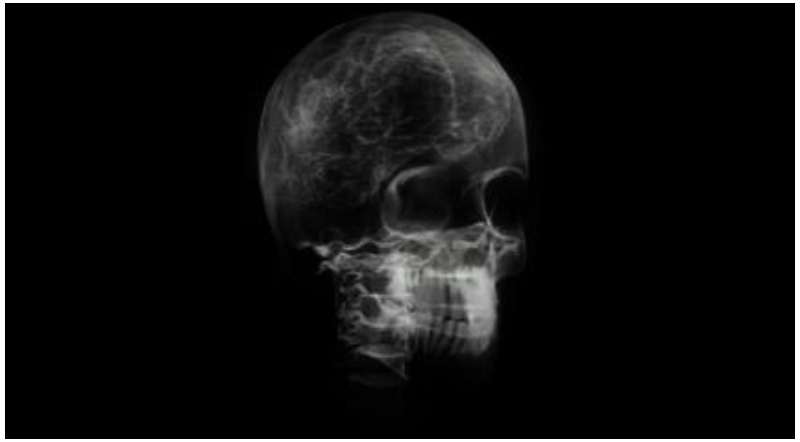






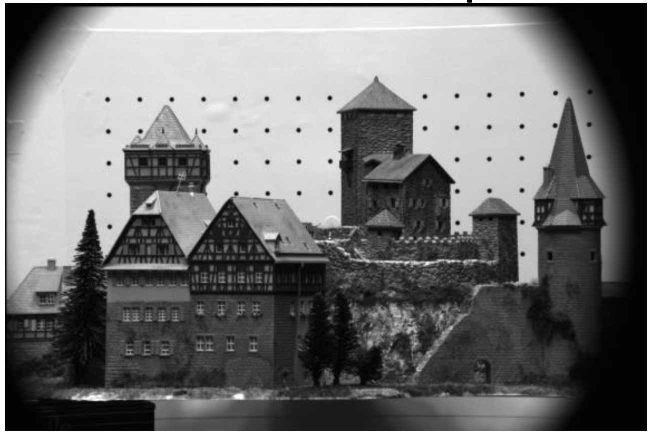






Sztereó látás és nehézségei

Bal kamera képe



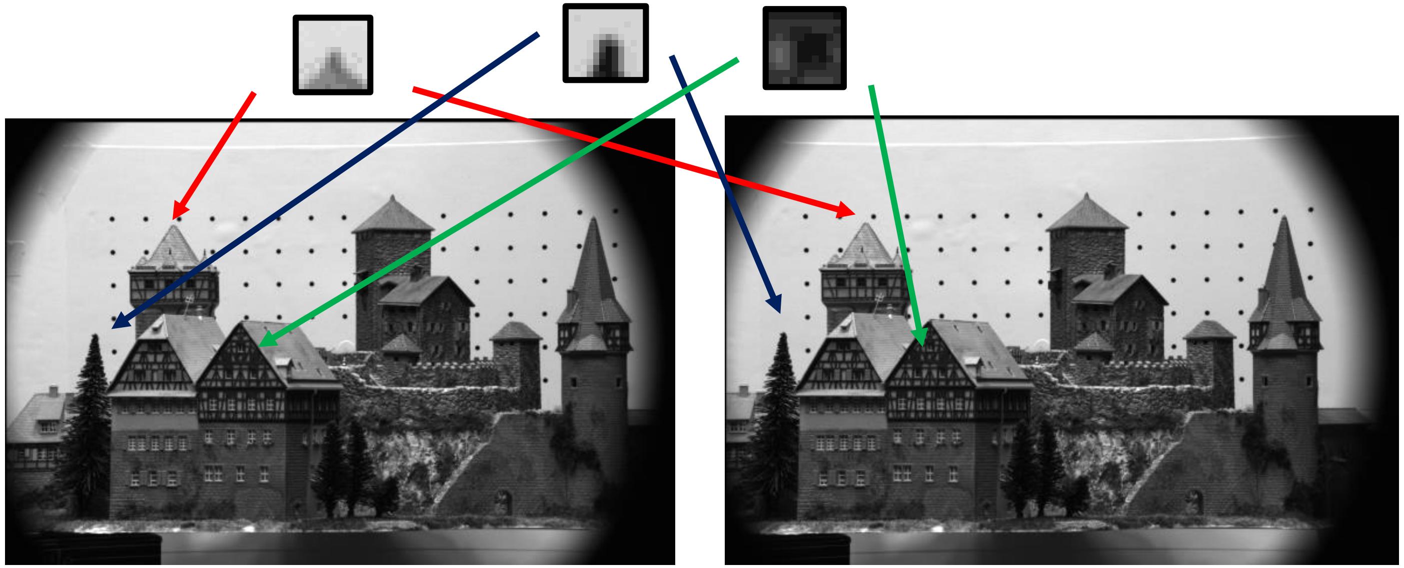
Jobb kamera képe

Elképzelhető, hogy

- Valami nem látszik mindkét képen
- A keresett pont takarásba kerül más objektumok által

Diszparitás probléma: Keressük meg mindkét képen az egymásnak megfelelő pontokat (képjellemzőket)!

Bal kamera képe



Jobb kamera képe

Megfeleltetési módszerek

1. Korrelációs módszerek
   - **Tömbök összehasonlításán alapul**

$$\boxed{c(dx,dy)=\sum_{k=-M}^{M}\sum_{l=-N}^{N}\psi\left(I_L(x_L+k,y_L+l),I_R(x_L+dx+k,y_L+dy+l)\right)}$$

     - Négyzetösszeg (Sum of square distance - SSD)

$$\Psi(u,v)=-(u-v)^2$$

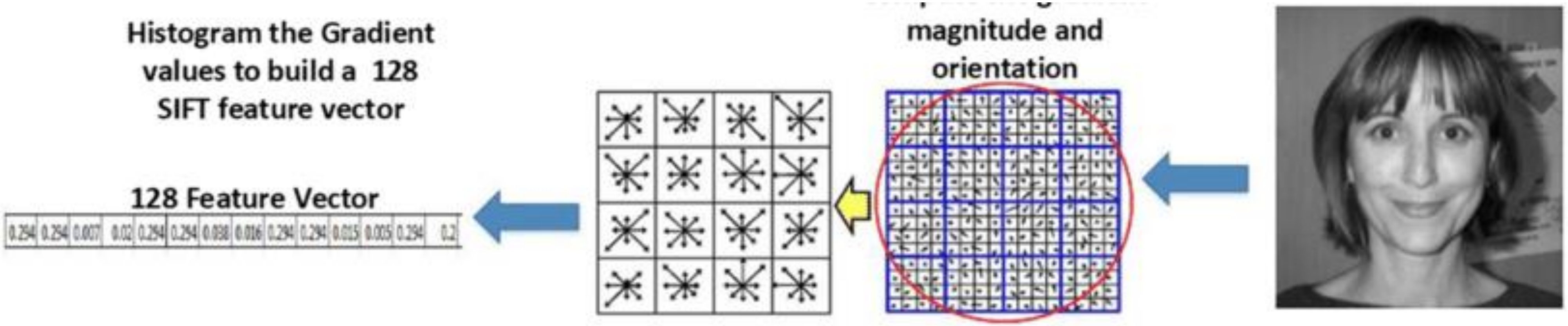
     - Abszolútérték-összeg (Sum of absolute distance – SAD)

$$\Psi(u,v)=-|u-v|$$

2. Képjellemzőkre alapuló módszerek
   - **SIFT (Skálázás invariáns képjellemző transzformáció)**

     (eltolás, elforgatás, átskálázás és kismértékű torzításra invariáns képleírókat)

     - Sarokpontok
     - Élek
     - Formák
     - …



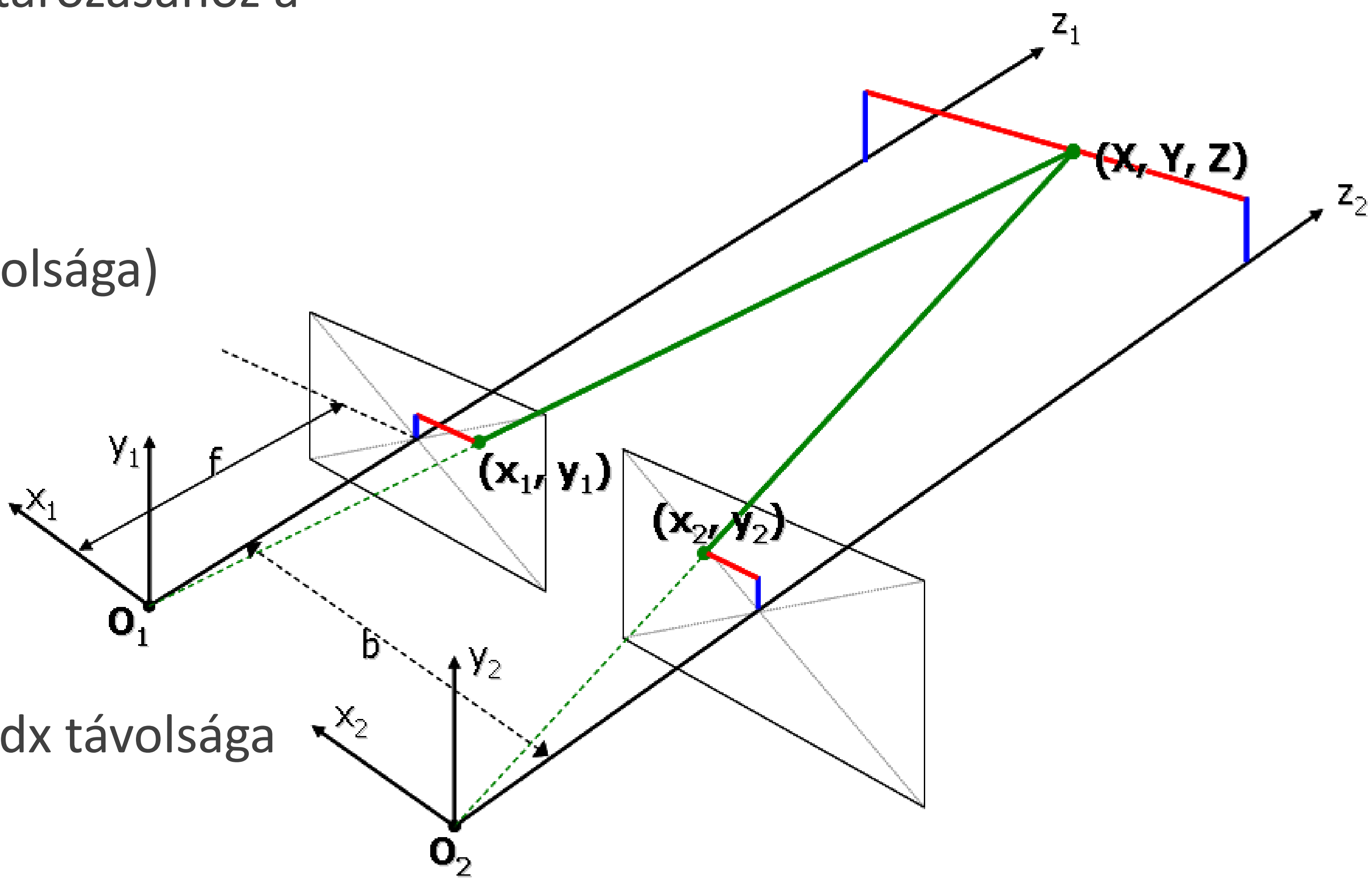
Sztereó látás matematikai alapjai

Párhuzamos optikai tengelyek esetén (koplanáris elrendezés)

A mélységi információ meghatározásához a kalibrációból származik:

- fókusztávolság
- bázistávolság (két kamera távolsága)

$$\boxed{Z = \frac{fb}{x_1 - x_2} = \frac{fb}{dx}}$$



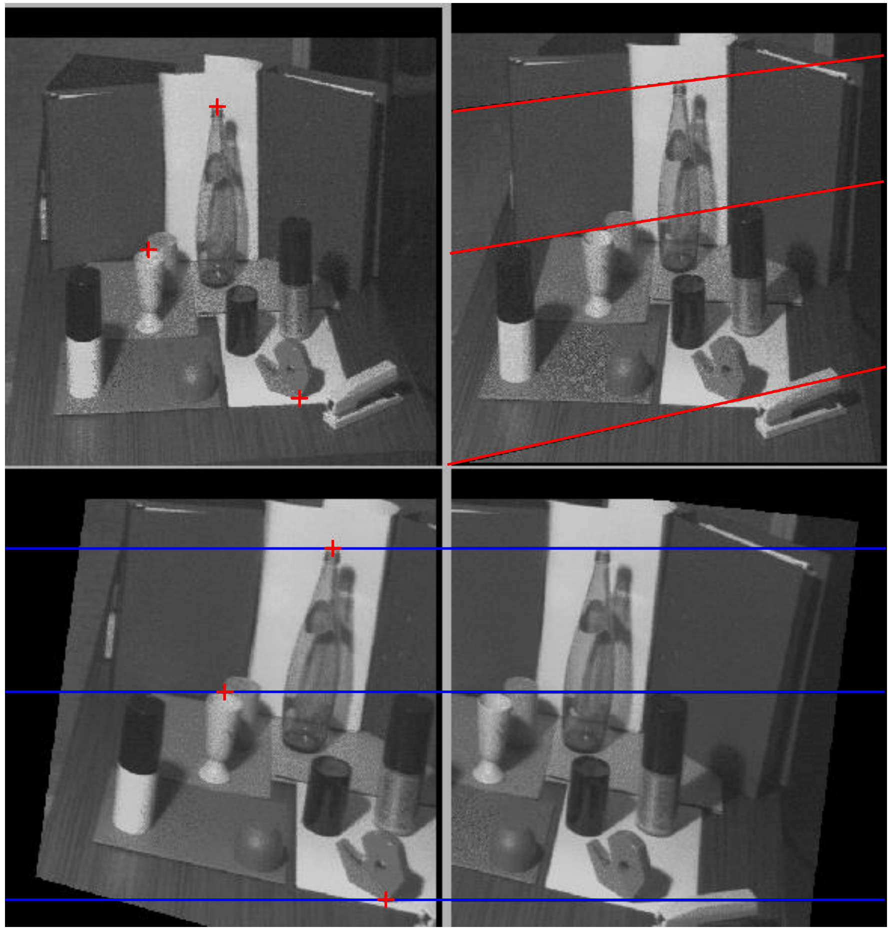
A képek alapján mérhető a

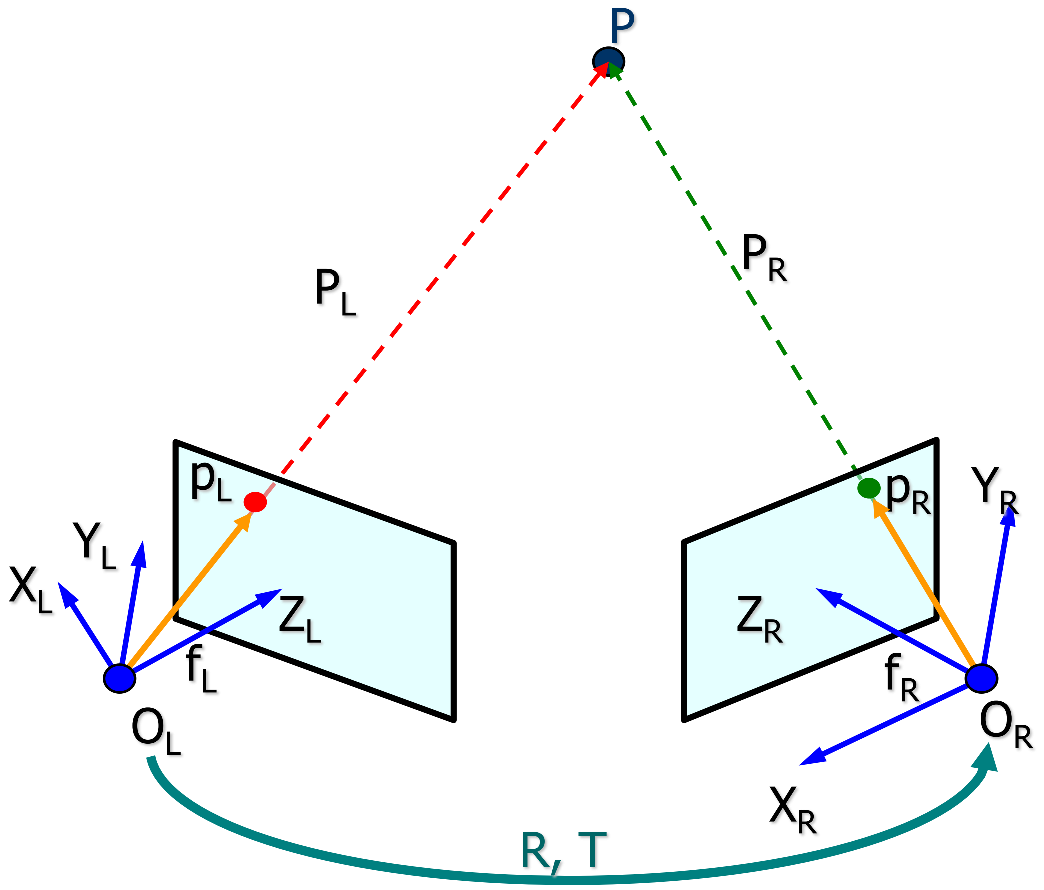
- megfelelő képpont eltolódás dx távolsága

Sztereó látás általános esetben

Rektifikálás: Képek torzítása úgy, hogy visszakapjuk a koplanáris elrendezést

Bal kamerakép Jobb kamerakép





Mozgásból származó információ felhasználása a képfeldolgozásban

Mozgás

Ha folyamatában nézzük a képeket, akkor **többletinformáció** nyerhető ki → a **mozgás** vizsgálata

A mozgás rengeteg információt hordoz magában:

- A perspektíván keresztül térérzetet illetve térbeli információt ad az objektumokról (motion parallax)
- Sebesség érzékelés (veszélyek felismerése)
- Mozgó objektum elkülönítése (objektumszegmentálás)
- Objektumfelismerés és azonosítás

(pl. viselkedés, mozgás alapján)

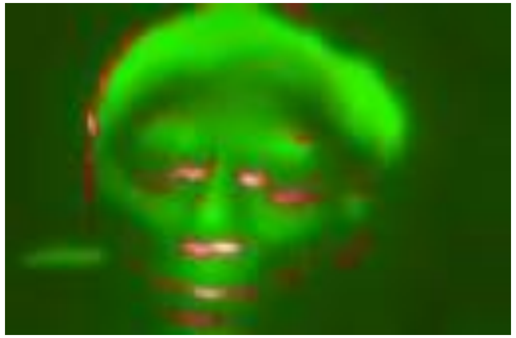
- …

Érdemes átemelni a számítógépes látórendszerekbe is ezt a tudást!

Mozgás észlelése az élőlényeknél

Már a retina szintjén megtörténik a képjellemzők és a mozgás szegmentálása, az agy tömörített információt kap!

animáció



On-Off irányszelektív sejtek, és az Off Sluggish sejtek tüzelése a retinában

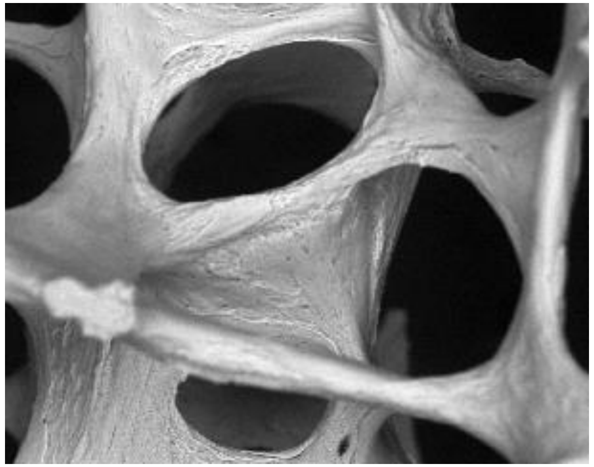
Mozgás alapú képfeldolgozás

Alkalmazások:

- Biztonságtechnika: „Valami mozog, figyelj oda!”
- Kameramozgás korrekciója: képstabilizálás
- Objektumkövetés (összefüggő régió követése)
- Mozgás alapú felismerés (egyéni sajátosságok)
- Mozaikozás (képek egymáshoz igazítása, pl. panoráma)
- Kameramozgás alapján 3D rekonstrukció
- Videó tömörítés (mpeg)
- …

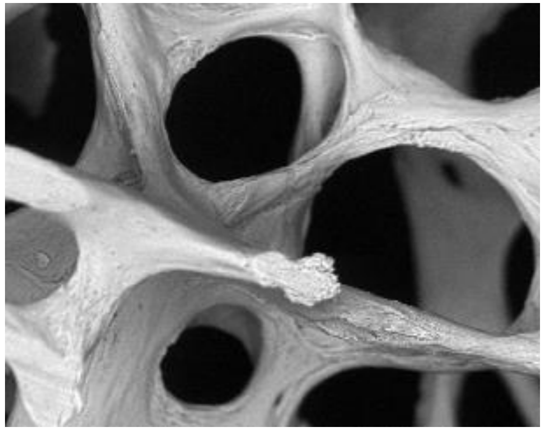
Egy korábbi példa (motion parallax)

Alakmeghatározás mozgás alapján (shape from motion). Ha az objektum a képsíkkal párhuzamosan mozog, nagyon hasonló a sztereó látáshoz, hiszen mindegy, hogy az objektum kis mértékben elmozdul vagy egy másik közeli pontból nézek rá.

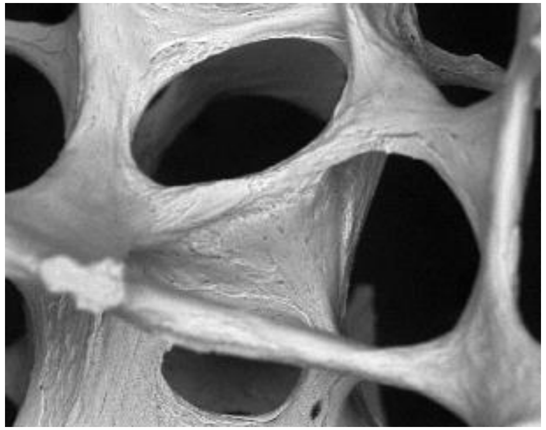


animáció

1. állapot



2. állapot



Mozgásparallaxis matematikai alapjai
(Motion stereo)

Mozgásparallaxis, mozgás sztereó

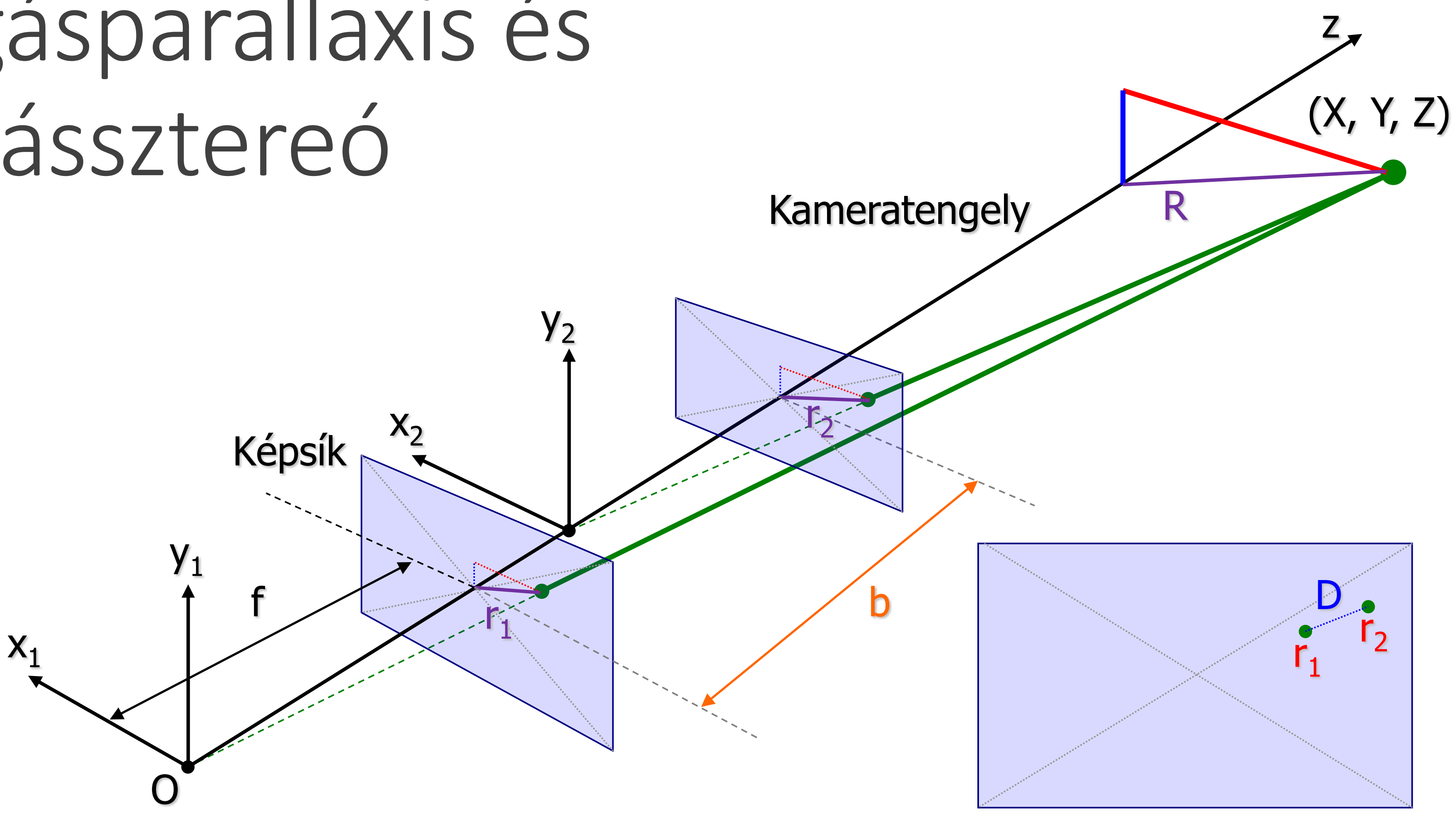
Monokuláris (egykamerás) elrendezés, több felvételt készítünk ugyanarról a jelenetről (mozgó térbeli pontról).
A megoldás a korábban tárgyalttal megegyezik.

Most **ne a sztereó diszparitást** (optikai tengelyre merőleges, képsíkkal párhuzamosan történő elmozdulást) vizsgáljuk, hanem az optikai tengellyel párhuzamos, tehát mélység irányú különbségeket nézzük!

Kérdés:
Hogyan számíthatjuk ki a mélységinformációt?

Megjegyzés: A „mozgássztereó (motion stereo)” és a „mozgásparallaxis (motion parallax)” kifejezések gyakorlatilag szinonimaként használhatók, mivel mindkét esetben két képsíkunk van, csak **az egyik esetben a két képsík egymás mellett, a másik esetben egyik a másik előtt helyezkedik el**.

Mozgásparallaxis és mozgássztereó

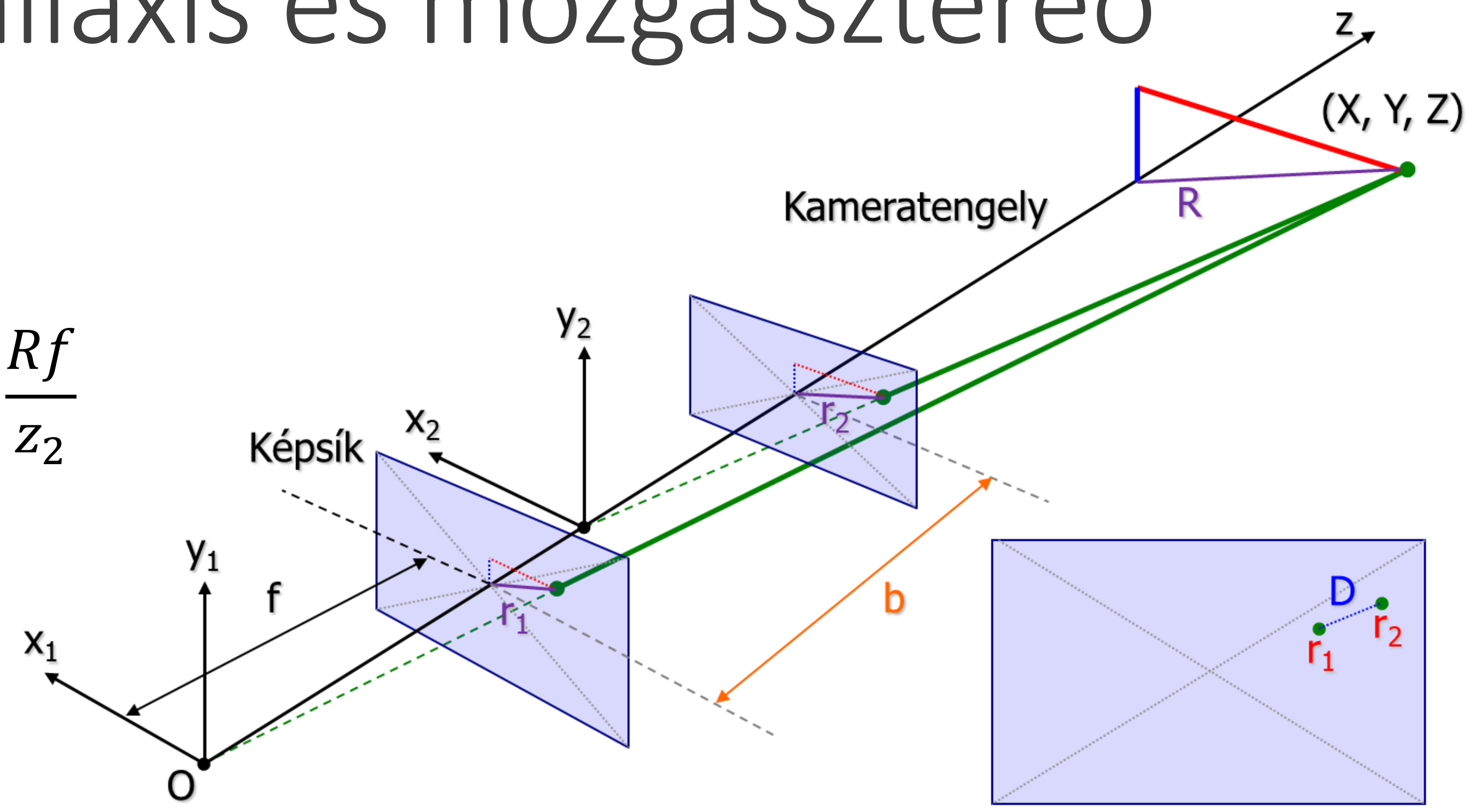


Elméletileg mindegy, hogy a tárgy mozog a kamera optikai tengelyével párhuzamosan, vagy a kamera mozdult el b távolsággal.

Mozgásparallaxis és mozgássztereó

$$R = \sqrt{(X^2 + Y^2)}$$

$$r_1 = \frac{Rf}{z_1} \qquad r_2 = \frac{Rf}{z_2}$$



A pont elmozdulása a képsíkon

Ismét a hasonló háromszögek elvét használtuk fel.
A vizsgált képpont elmozdulása (képeken mért távolsága) kifejezhető:

$$D = r_1 - r_2 = Rf\left(\frac{1}{z_1} - \frac{1}{z_2}\right)$$

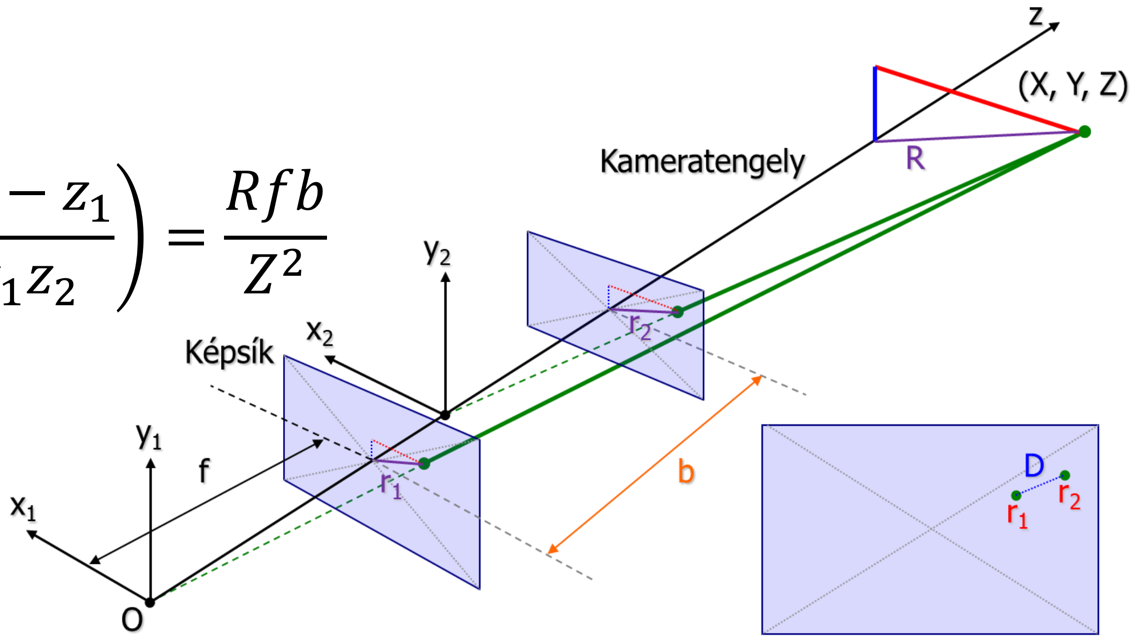
Mozgásparallaxis, mozgássztereó

Ha a bázistávolság $\boxed{b = z2 - z1}$ és
b << z1, z2, akkor

$$D = Rf\left(\frac{1}{z_1} - \frac{1}{z_2}\right) = Rf\left(\frac{z_2 - z_1}{z_1 z_2}\right) = \frac{Rfb}{Z^2}$$

Az ábrából kiolvasható, hogy

$$\frac{R}{Z} = \frac{r}{f}, \text{ahol } r \approx \frac{r_1 + r_2}{2}$$



Így:

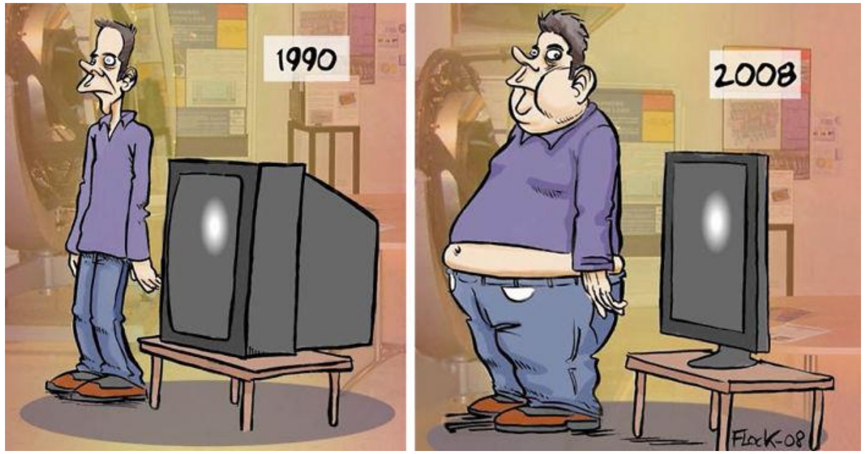
$$D = \frac{Rfb}{Z^2} = \frac{R}{Z}\frac{fb}{Z} = \frac{r}{f}\frac{fb}{Z} \Rightarrow \boxed{Z = \frac{br}{D}}$$

Tehát ha ismert a kamera vagy a tárgy tengelyirányú elmozdulása (b), a képpont távolsága a középponttól (r), illetve a képpont (nem feltétlenül vízszintes) elmozdulása a képen (D), akkor kiszámolható a tárgypont távolsága (Z).

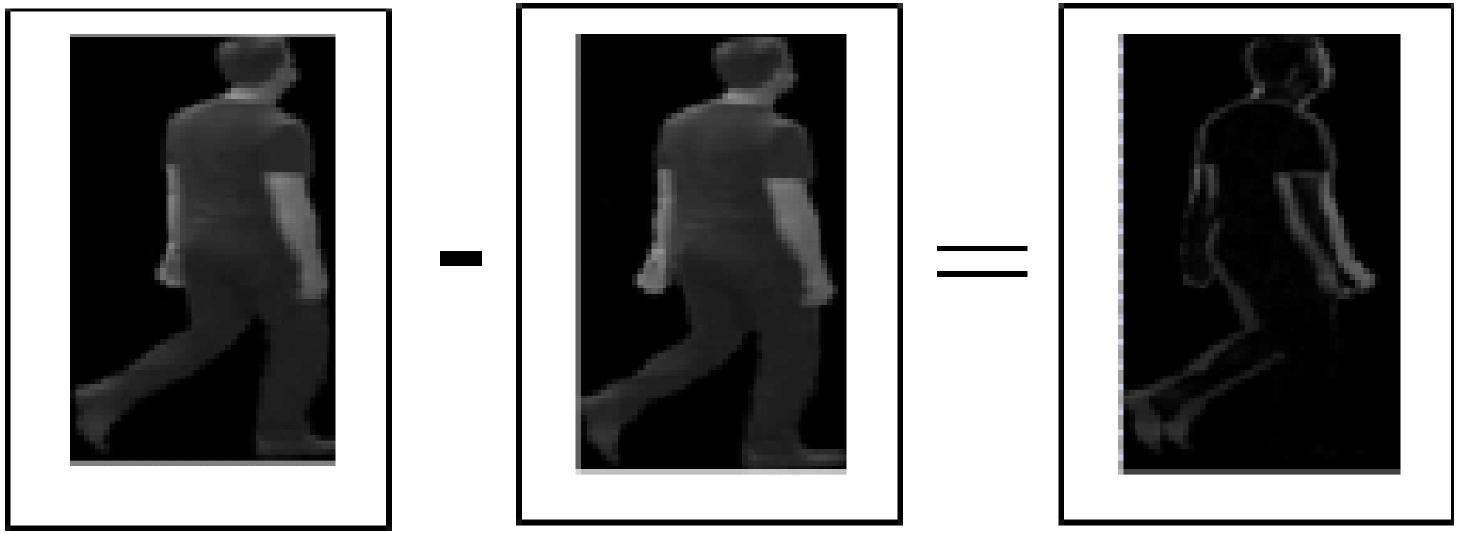
Mozgásérzékelés differenciaképek segítségével

Mozgásfelismerés

Képezzük a különböző időpontban
készült képek különbségét!



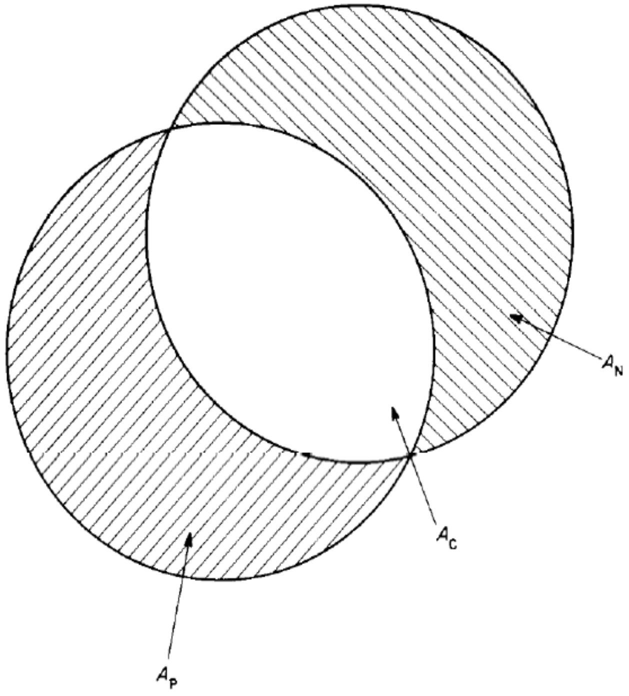
A cél a mozgó régió ROI (Region of Interest) kijelölése, elkülönítése, ami a további feldolgozást (pl. felismerést) hatékonyabbá teheti.



A zaj miatt akár a teljes kép változhat, így a ROI a teljes kép lesz!

Akkumulatív differencia képek

A szomszédos frame-ek pixelein a differencia képet a céljainknak megfelelően többféleképpen vizsgálhatjuk.

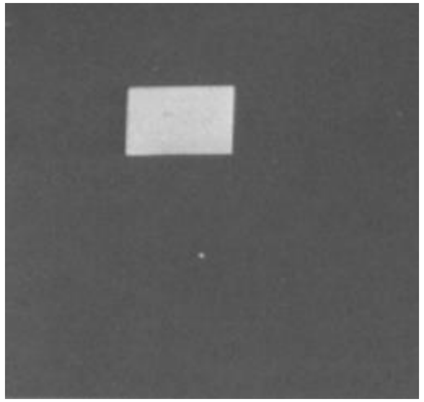


Definiáljuk a pozitív akkumulatív differencia (PDP), a negatív akkumulatív differencia (NDP), és az abszolút akkumulatív differencia (ADP) fogalmat két egymás után felvett képen:

- $\mathrm{PDP}(x,y) = \begin{cases} 1, \text{ha } F_1(x,y) - F_2(x,y) > T_P \\ 0, \text{egyébként} \end{cases}$
- $\mathrm{NDP}(x,y) = \begin{cases} 1, \text{ha } F_1(x,y) - F_2(x,y) < T_N \\ 0, \text{egyébként} \end{cases}$
- $\mathrm{ADP}(x,y) = \begin{cases} 1, \text{ha } |F_1(x,y) - F_2(x,y)| > T_A \\ 0, \text{egyébként} \end{cases}$

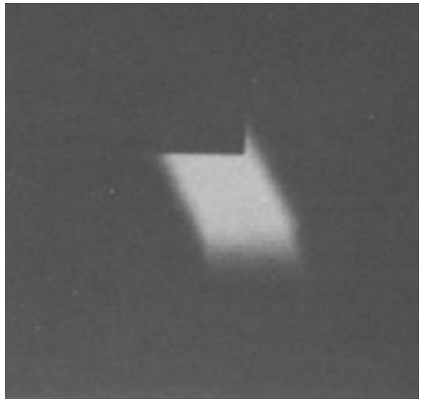
A képsorozatok akkumulatív differencia képei

Például homogén objektum egyirányú mozgása



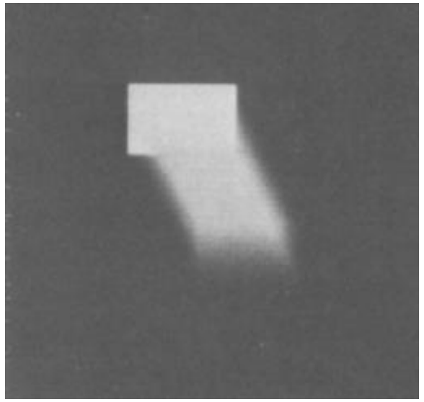
PDP

nagy intenzitású objektum eltűnt



NDP

látszik az objektum „csóvája”



ADP

bármilyen küszöb feletti változás

Zajszűréssel egybekötött mozgásérzékelés

LoG (Laplacian of Gaussian) szűrő képtartományban (emlékeztető)

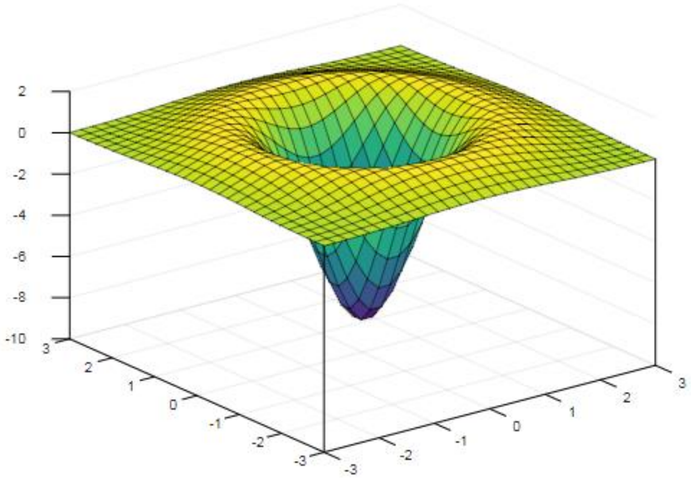
Zajszűrés és élkeresés egyben

σ:=1,4 (ez egy változtatható paraméter)

A konvolúciós ablak:

$$LoG\_5x5 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 \\ 1 & 2 & -16 & 2 & 1 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

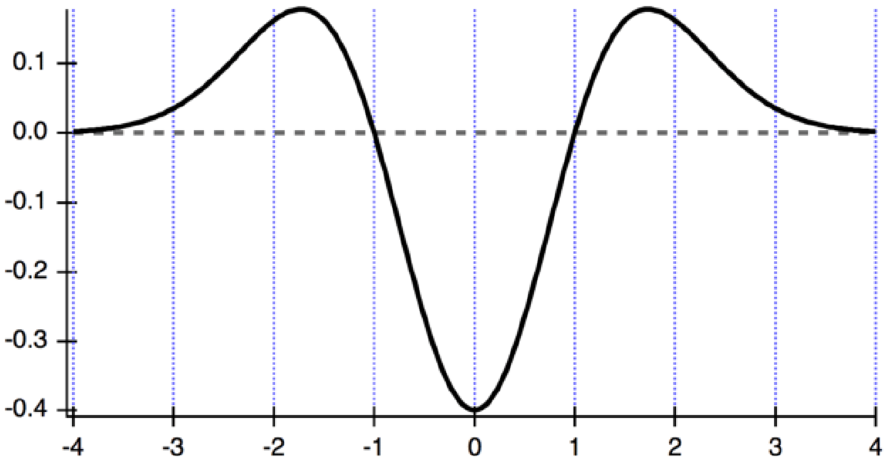
$$LoG\_9x9 = \begin{bmatrix} 0 & 0 & 1 & 2 & 2 & 2 & 1 & 0 & 0 \\ 0 & 1 & 3 & 5 & 5 & 5 & 3 & 1 & 0 \\ 1 & 3 & 5 & 3 & 0 & 3 & 5 & 3 & 1 \\ 2 & 5 & 3 & -12 & -23 & -12 & 3 & 5 & 2 \\ 2 & 5 & 0 & -23 & -40 & -23 & 0 & 5 & 2 \\ 2 & 5 & 3 & -12 & -23 & -12 & 3 & 5 & 2 \\ 1 & 3 & 5 & 3 & 0 & 3 & 5 & 3 & 1 \\ 0 & 1 & 3 & 5 & 5 & 5 & 3 & 1 & 0 \\ 0 & 0 & 1 & 2 & 2 & 2 & 1 & 0 & 0 \end{bmatrix}$$

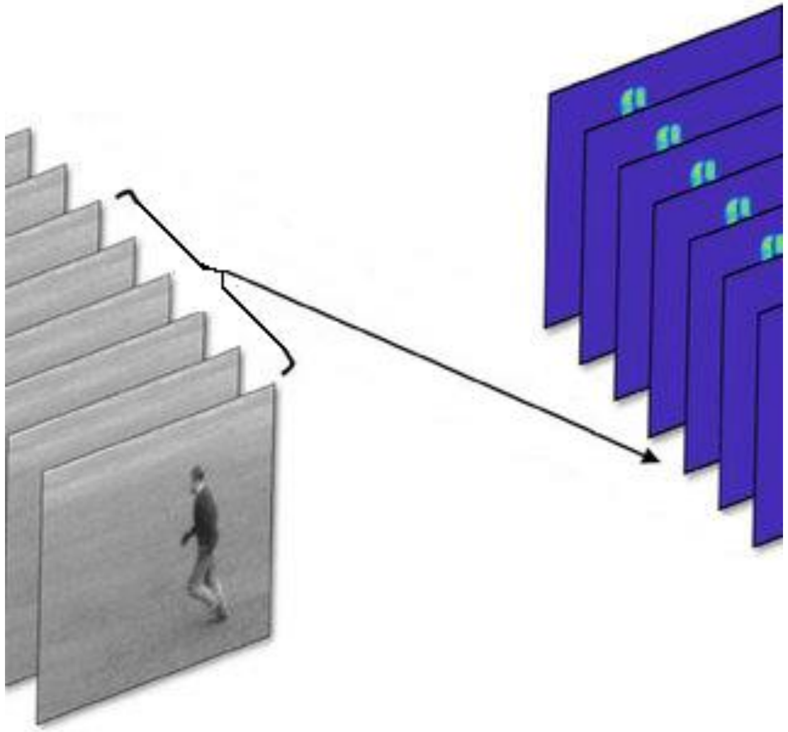


LoG szűrő időtartományban

Alkalmazzuk az LoG szűrőt időtartományban úgy, hogy az adott pixelnek egy időablakban felvett értékeit súlyozzuk vele.

Ezzel egyszerre oldjuk meg a pixelek termikus zajának a csökkentését (Gauss) és a drasztikus intenzitásváltozás (mozgás) kiemelést (Laplace).

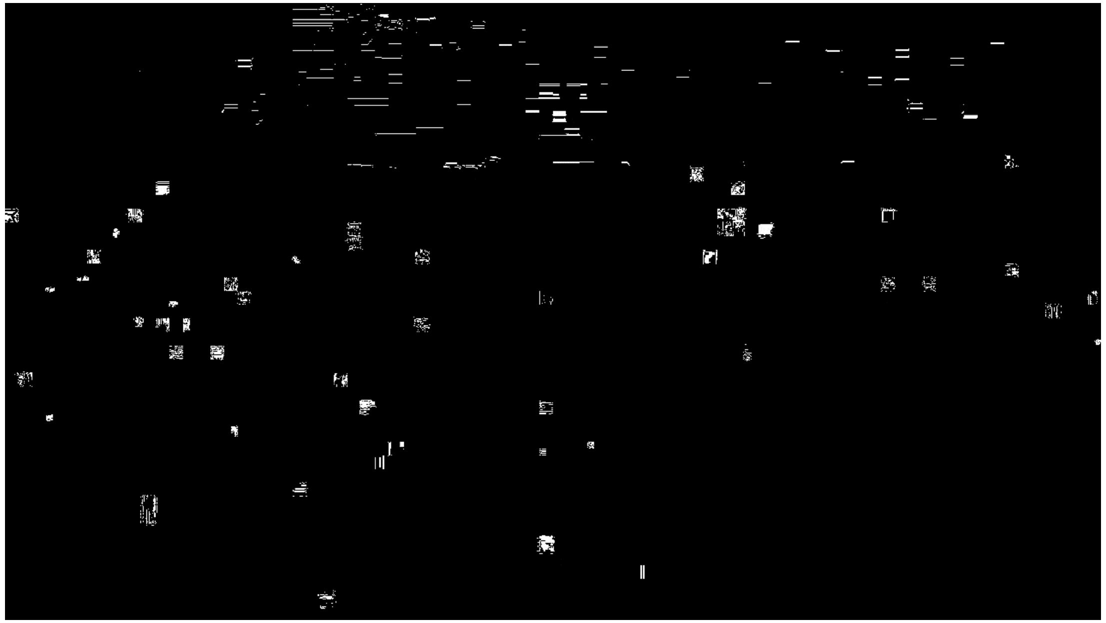




Morfológiai zajszűrés (ld. következő előadás)

A képen a termikus és pixelszerű zajok miatt kisméretű foltok keletkeznek.

A következő előadáson látunk módszereket ezeknek a kisméretű foltoknak az eltüntetésére.

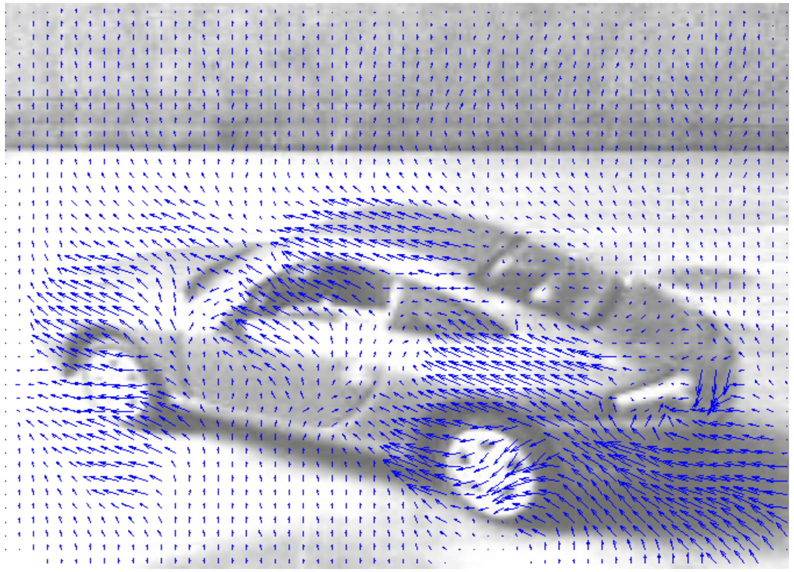


Optical flow (optikai áramlás)

A mozgás mesterséges ábrázolása

t.

t+1.



Nyilak mutatják a képrészlet elmozdulásának irányát és nagyságát

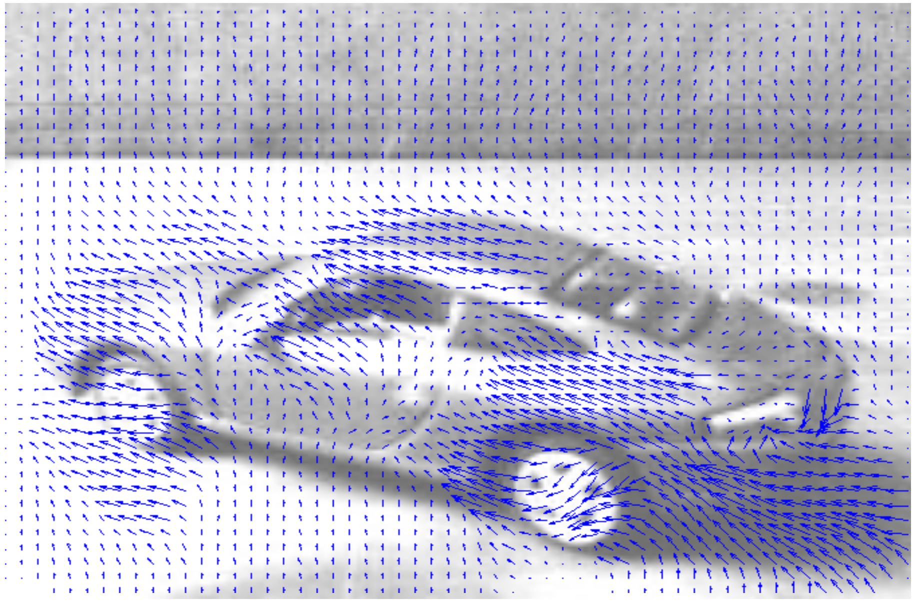
Az optical flow definíciója

Az optikai áramlás egy vizuális jelenetben a megfigyelő és az objektumok relatív elmozdulása által létrejött látszólagos intenzitásmintázat változás.

Képpontokhoz vektor hozzárendelése → megmutatja, hogyan jutunk el a következő képkockához

vektormező → optical flow (mező)

Bár sokan nem teljesen helyesen **elmozdulásmező**nek hívják, de igazából **sebességmező**! Hol van jelen a képben az idő a sebesség interpretációhoz?



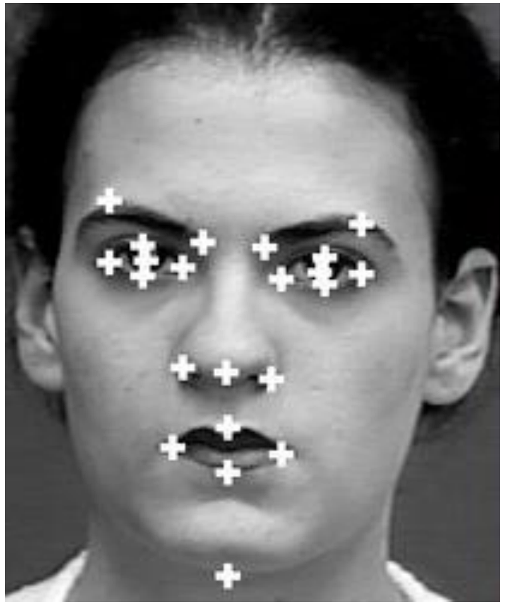
Objektumkövetés stratégiák

1. Az objektumon lévő jellegzetes pontok (pl. sarkok) alapján
   Feature Tracking

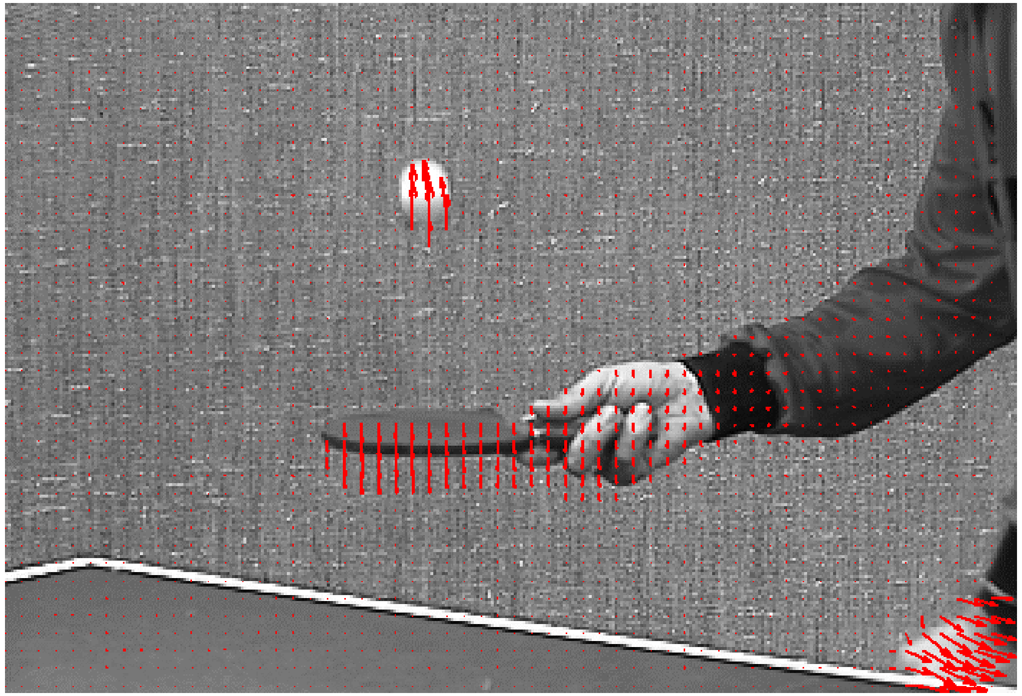
   Előnye, hogy csak a jellegzetes pontokat kell követni → kisebb számítási igény
2. A teljes képen a textúraelemek elmozdulása alapján
   Optical Flow

   Előnye, hogy homogén számítási jellege van a teljes képre (például GPU-n hatékonyabb)

A FT interpretációs ereje nagyobb, a jellegzetes pontok önálló jelentést hordozhatnak (pl. szájzug).



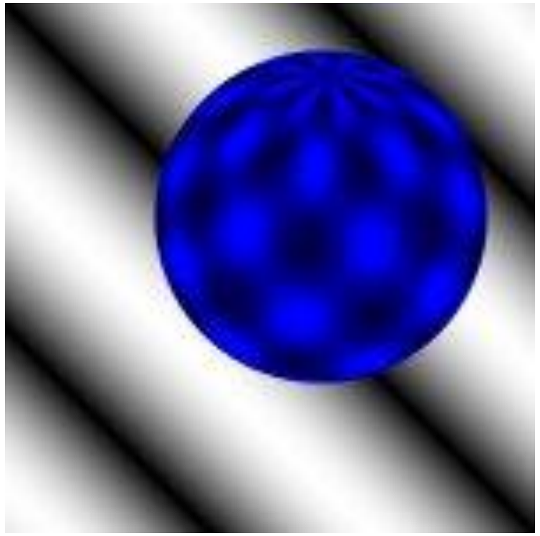
Optical flow alkalmazás: Mozgás alapú szegmentálás

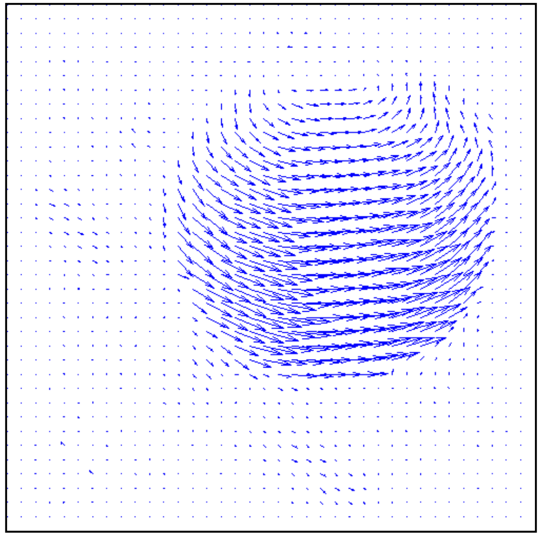


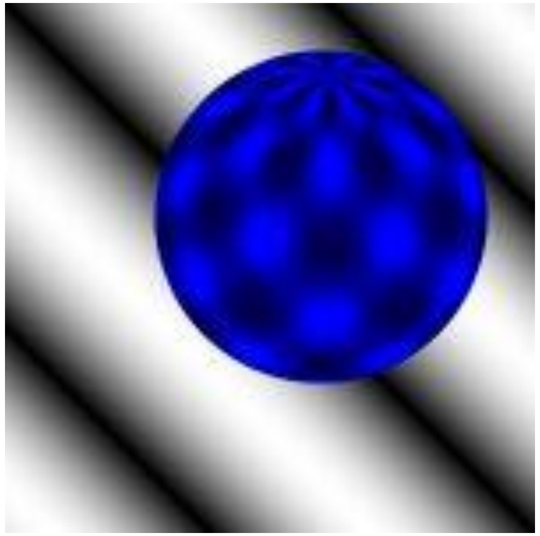
Figyelem! Bizonyíthatóan digitálisan manipulált videó!

Sebességmező nehézségek

- Homogén mező mozgásakor mit tapasztalunk?
- A gömb minden pontja ugyanakkora sebességgel forog! Látható, hogy a sebességmező nem a valódi sebességet mutatja! Mi okozza az eltérést?
- Később még lesz néhány példa a nehézségekre.

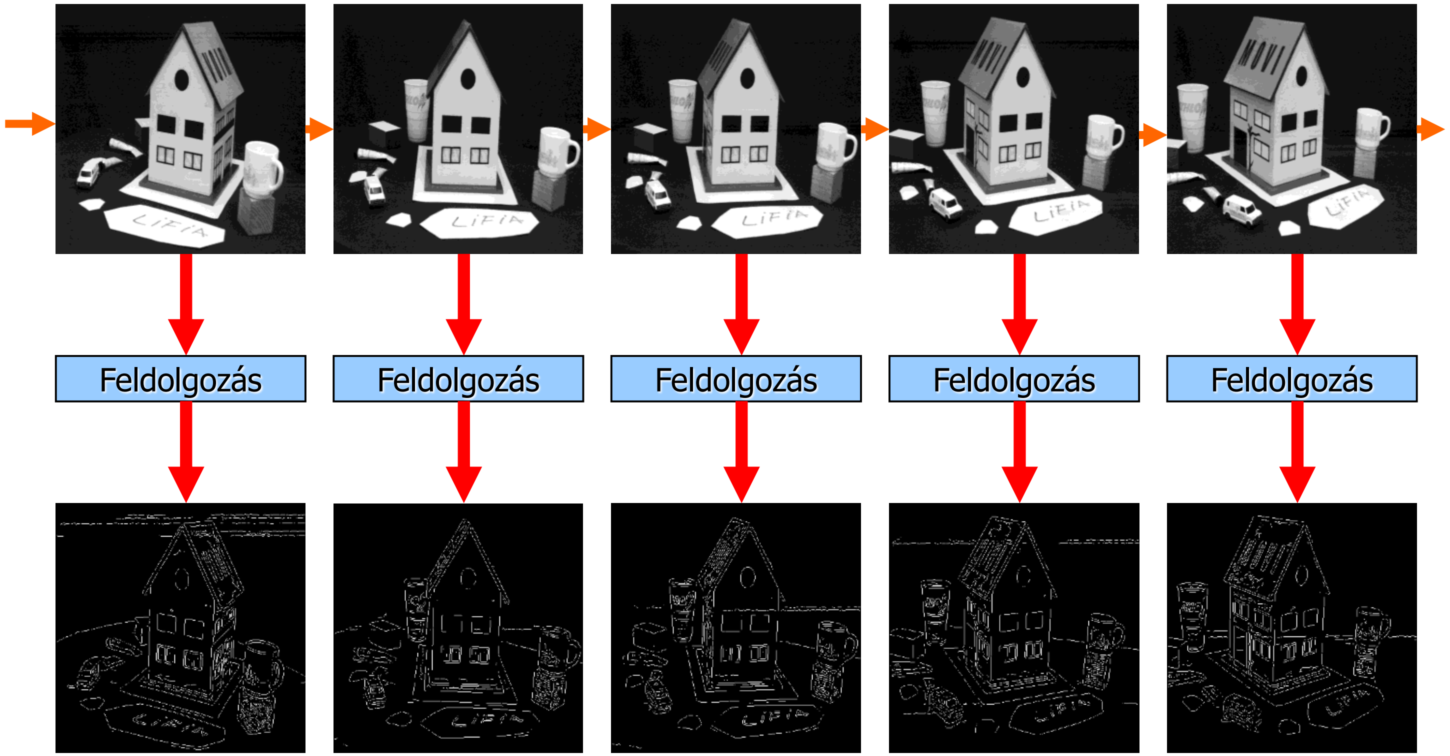






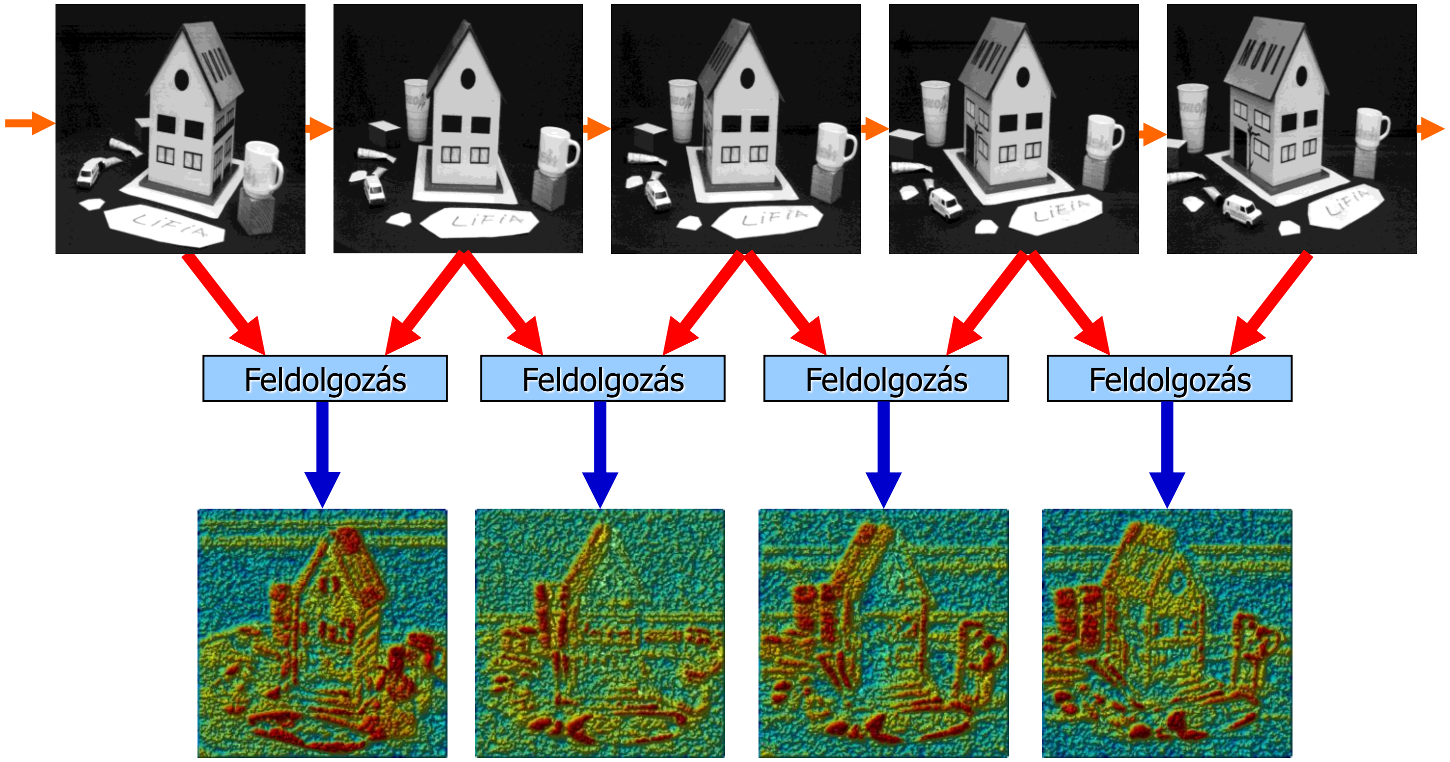
Képfeldolgozási stratégiák

Független feldolgozás: képek vagy képrészletek egymástól függetlenül történő feldolgozása



Képfeldolgozási stratégiák

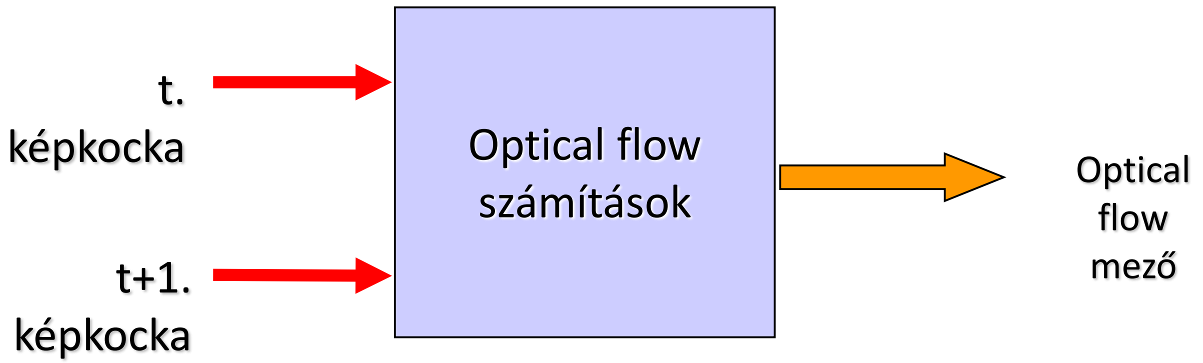
Az optical flow esetén erősebb függés nem csak a képrészletek, hanem a szomszédos képek között is. (Párhuzamosíthatóság?)



Optical flow feladat összefoglalása

Bemenet: két, egymást követő képkocka

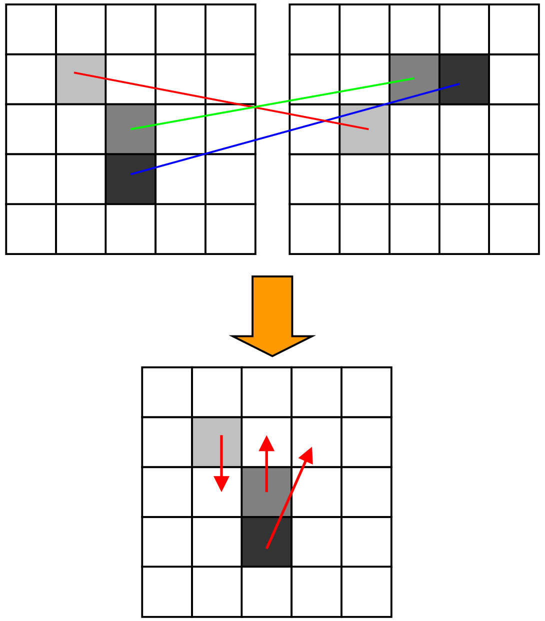
Kimenet: optical flow mező



Elmozdulás számítás

Összetartozó pixelek: ugyanannak a térbeli pontnak a két projekciója

Kérdés: Hogyan lehet az összetartozó pixeleket megtalálni?



Hogyan lehet egy pixel elmozdulását kiszámítani?

- A t. képen egy pixel párját megkeressük a t+1. képen
- A helyzetek különbsége lesz az elmozdulás

Optical flow feltételezések

- Egyelőre egyetlen kamerával dolgozunk
- Tegyük fel, hogy a háromdimenziós térben történő mozgás leképzése a képsíkon is mozgást eredményez.
- Az információ szinte mindig redukálódik!
- Legalább két kép kell a meghatározásához (a panorámaképnél láttuk, hogy a mozgás egy képen is megjelenhet, de abban az esetben is több pillanatot rögzítettünk)
- A képrészlet két képkocka között nagyon kis mértékben mozdul el Δx és Δy irányban, miközben intenzitása legfeljebb nagyon kis mértékben változik.

$$E(x, y, t) \approx E(x + \Delta x, y + \Delta y, t + \Delta t)$$

Állandó intenzitás leírása 1.

Rendelkezésre álló adatok:

- A mozgó pont helyzete
  $x(t), y(t)$ → a t. időpillanatban a pont koordinátái
- A pixel leírása:
  - $E(x, y, t)$ → a t. időpillanatban az (x, y) pixel intenzitását adja meg

A mozgó pont intenzitása a t. időpillanatban:

$$E(x(t), y(t), t)$$

Állandó intenzitás leírása 2.

A mozgó pont intenzitása állandó:

$$E\big(x(t+\Delta t), y(t+\Delta t), t+\Delta t\big) = E\big(x(t), y(t), t\big)$$

Írjuk fel az egyenlet Taylor-sorát:

$$E(x,y,t) \approx E(x+\Delta x, y+\Delta y, t+\Delta t) =$$
$$= E(x,y,t) + \frac{\partial E}{\partial x}\cdot\Delta x + \frac{\partial E}{\partial y}\cdot\Delta y + \frac{\partial E}{\partial t}\cdot\Delta t + \text{magasabb rendű, elhanyagolt tagok}$$

A magasabb rendű tagokat azért hanyagoljuk el, mert két képkocka között csak kis mértékű, lineáris változást feltételezünk.

Az egyenlet mindkét oldalából levonható a képrészlet eredeti helyen, a korábbi képkockán mért intenzitása:

$$\frac{\partial E}{\partial x}\cdot\Delta x + \frac{\partial E}{\partial y}\cdot\Delta y + \frac{\partial E}{\partial t}\cdot\Delta t \approx 0$$

Az optical flow korlátozás

A fenti egyenletet leosztva a Δt idővel

$$\frac{\partial E}{\partial x} \cdot \frac{\Delta x}{\Delta t} + \frac{\partial E}{\partial y} \cdot \frac{\Delta y}{\Delta t} + \frac{\partial E}{\partial t} \approx 0$$

Nagyon kicsi változásokat feltételezve adódik az **optical flow korlátozás** egyenlete:

Gradiens vektor x komponense

Gradiens vektor y komponense

Kép időbeli változása

$$\frac{\partial E}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial E}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial E}{\partial t} = 0$$

x irányú sebesség

y irányú sebesség

Az optical flow alapegyenlet

Bevezetve az (u, v) sebességvektorokat, kapjuk az optical flow alapegyenletét:

$$\frac{\partial E}{\partial x} \cdot u + \frac{\partial E}{\partial y} \cdot v + \frac{\partial E}{\partial t} \approx 0$$

Vagy egyszerűbb alakokban:

$$\boxed{E_x \cdot u + E_y \cdot v + E_t \approx 0}$$

vektoros alak:

$$\nabla E(\mathbf{x}, t) \cdot \mathbf{v} + E_t(\mathbf{x}, t) = 0$$

Az egyenlet értelmezése:

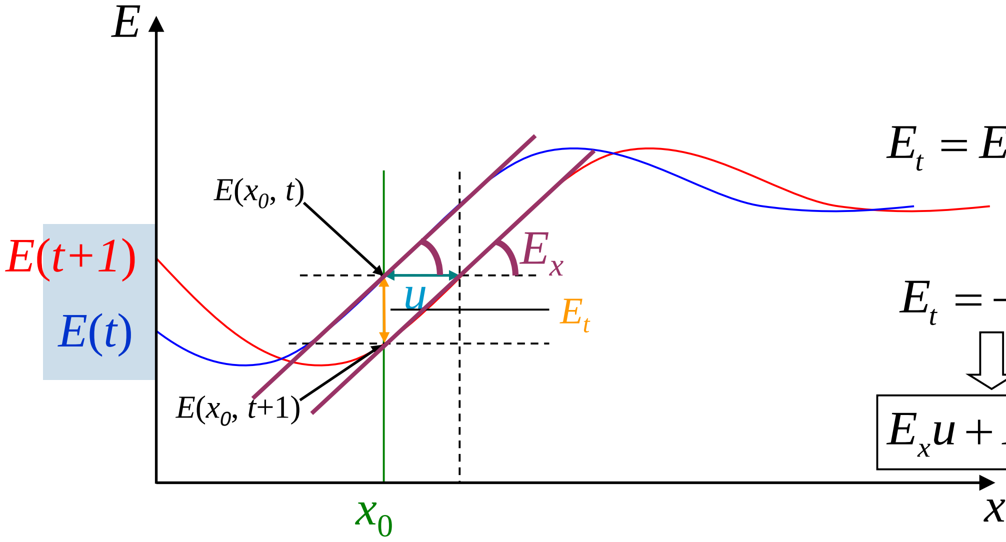
A változást 3 komponensre (tagra) bontottuk fel:

- Az intenzitás x irányú megváltozása (képgradiens x irányú komponense) * x irányú u sebességvektor
- Az intenzitás y irányú megváltozása (képgradiens y irányú komponense) * y irányú v sebességvektor
- Az intenzitás időben történő kismértékű változása

Geometriai értelmezés

t. időpillanatban az (x, y) pont körüli területet egy olyan 3D felülettel lehet jellemezni, amelynek a meredeksége x irányban E_x, y irányban E_y

Ha a felszín a **t+1**. időpillanatig (u, v)-t mozdul el, akkor a fényesség változása E_t az (x, y) pontban



$$E_t = E(x, t+1) - E(x, t)$$

$$E_t = -E_x u$$

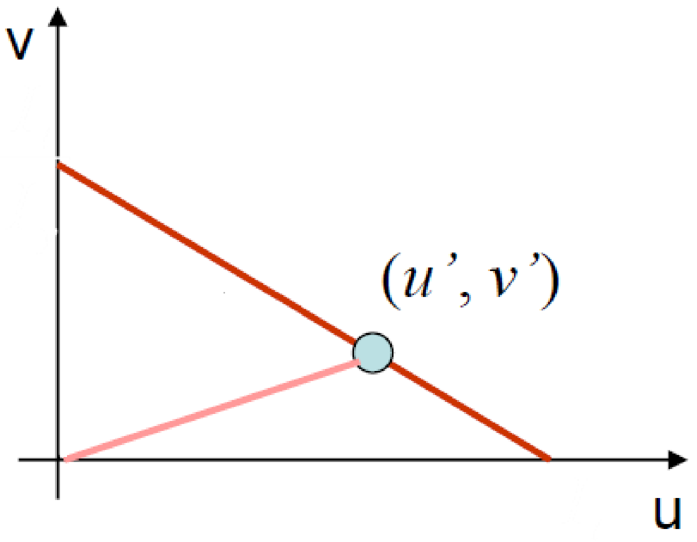
$$\Downarrow$$

$$\boxed{E_x u + E_t = 0}$$

Optical flow egyenlet problémája

Az optical flow alapegyenlete v és u szerint nézve egy lineáris egyenlet

$$I_x \cdot u + I_y \cdot v + I_t \approx 0$$



$$v = -u \cdot \frac{I_x}{I_y} - \frac{I_t}{I_y}$$

egyenes egyenlet

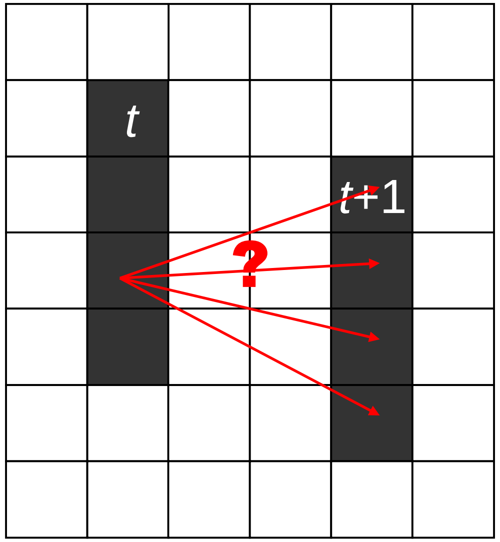
Látható, hogy az egyenes összes pontja megoldja az egyenletet (végtelen darab), de valójában csak egyetlen pont adja a tényleges megoldást, azt a sebességet, amivel a képrészlet elmozdult: (u', v')

Optical flow egyenlet problémájának következménye

Két ismeretlen (u és v), de csak egy egyenlet → nincs egyértelmű megoldás

Csak a **gradiensvektorral párhuzamos** irányú elmozdulást lehet kiszámítani

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

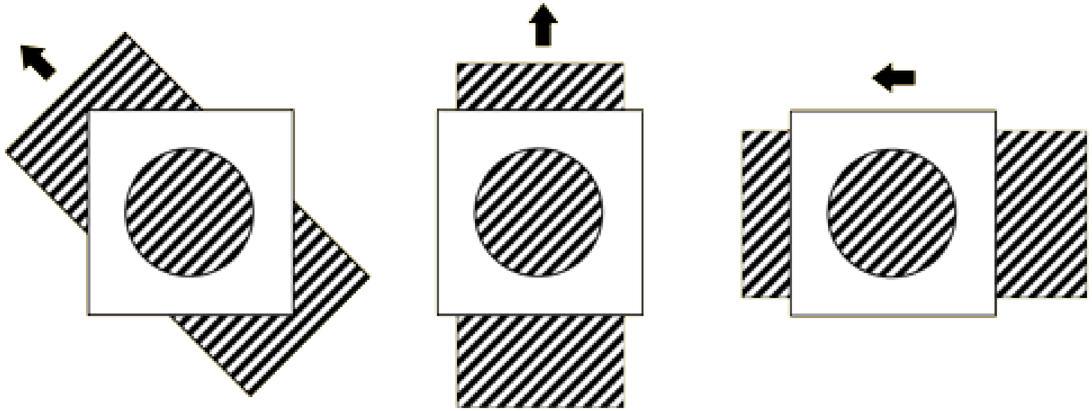


Apertúra probléma

Egy képrészlet (apertúra) alapján egyes esetekben a mozgásirányt nem tudjuk egyértelműen meghatározni.

Az oka az, hogy az optical flow alapegyenlet alulhatározott, ezért további **korlátozásokat kell bevezetni**.

„Barberpole" illúzió animáció



Az optical flow korlátozás kiegészítése

Szinte minden módszer alapját – közvetlenül, vagy közvetetten az optical flow korlátozás adja

Különbségek → hogyan egészítik ki a hiányos egyenletet, hogy egyértelmű legyen a megoldás

Különböző típusú technikák:

- Differenciális módszerek
- Korrelációs technikák
- Spektrumképre épülő módszerek

Differenciális technikák

Legrégibb módszerek

Nagyon jó minőség → az újabb *klasszikus* technikáknak sem sikerült érdemi javulást hozniuk

Sok egyéb technika alapját képezik

Legjelentősebb képviselői:

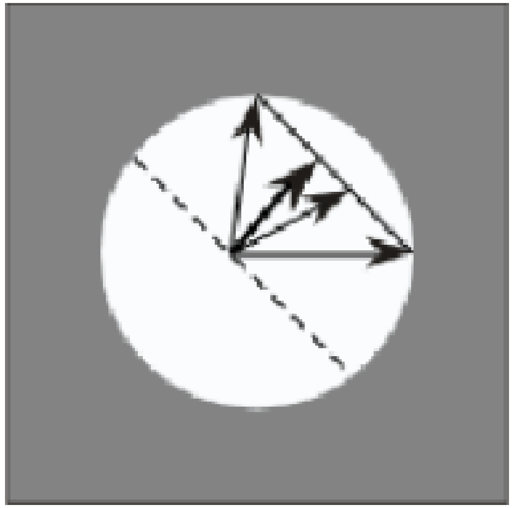
- Horn és Schunck módszere
- Lucas és Kanade módszere

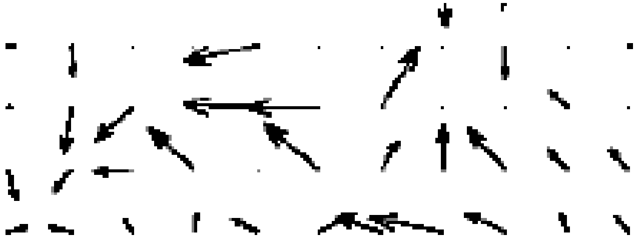
Horn és Schunck módszere

A legelső optical flow számító módszer (1980-1981)

Neveikhez fűződik az alapok lefektetése (optical flow fogalma, konstans intenzitás kényszere, optical flow korlátozás levezetése stb.)

Az ún. **egyenletességi korlátozással** (smoothness constraint) egészítették ki az optical flow kényszeregyenletet





Nem egyenletes vektormező

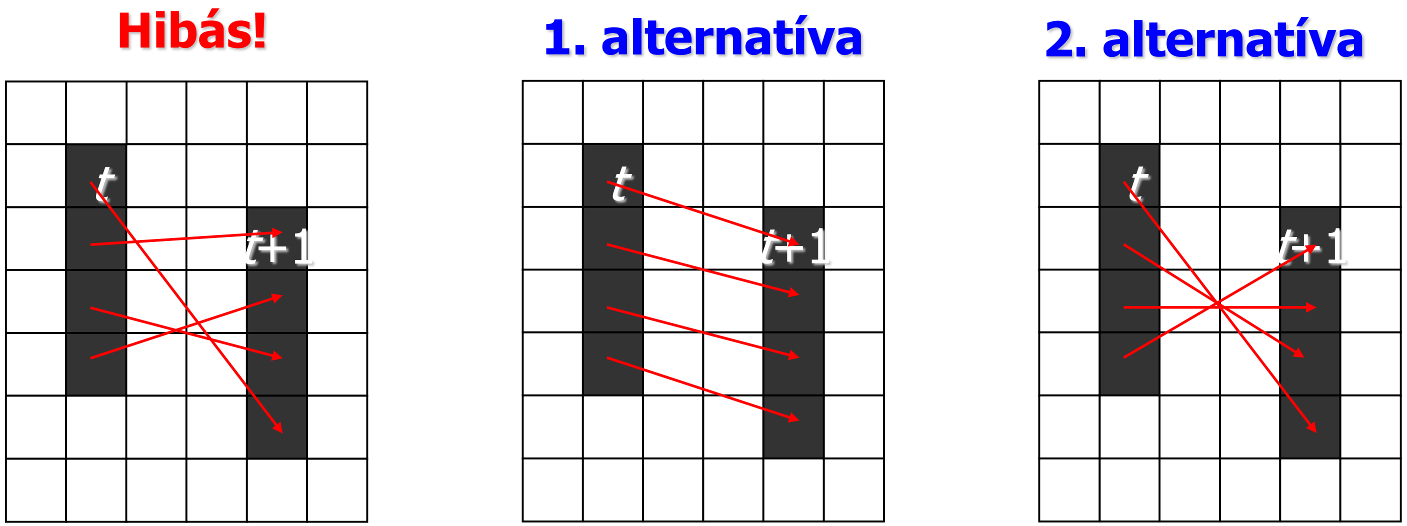
Egyenletes vektormező

Egyenletességi korlátozás

Feltételezi, hogy a szomszédos pontok **közel azonos** sebességgel mozognak → **a képsíkon a sebességvektorok egyenletesen változnak**

Vagyis egy kis területen nincsen hirtelen ugrás a sebességek változásában

Az objektumok határát (ahol lehet hirtelen változás) nem kezeli (!)



Egyenletességi korlátozás formálisan

Sebességvektorok változása:

$$\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}, \frac{\partial v}{\partial x}, \frac{\partial v}{\partial y}$$

A vektorok összes változása:

$$\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2$$

Az optical flow egyenlet kiszámítása

Vezessük be az optical flow korlátozás mellé az egyenletességi korlátozást, tehát vizsgáljuk azt, hogy egy kis területen egyenletes-e a sebességmező.

Az optical flow egyenlet egy újabb taggal bővül, ami a sebességekben lévő változásokat összegezi α súllyal:

$$\varepsilon_b = E_x u + E_y v + E_t$$

$$\varepsilon_c^2 = \left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2$$

$$\varepsilon^2 = \iint_D \left(\varepsilon_b^2 + \alpha^2 \varepsilon_c^2\right) dxdy$$

A feladat tehát ε^2 minimalizálása, amire vannak numerikus módszerek.

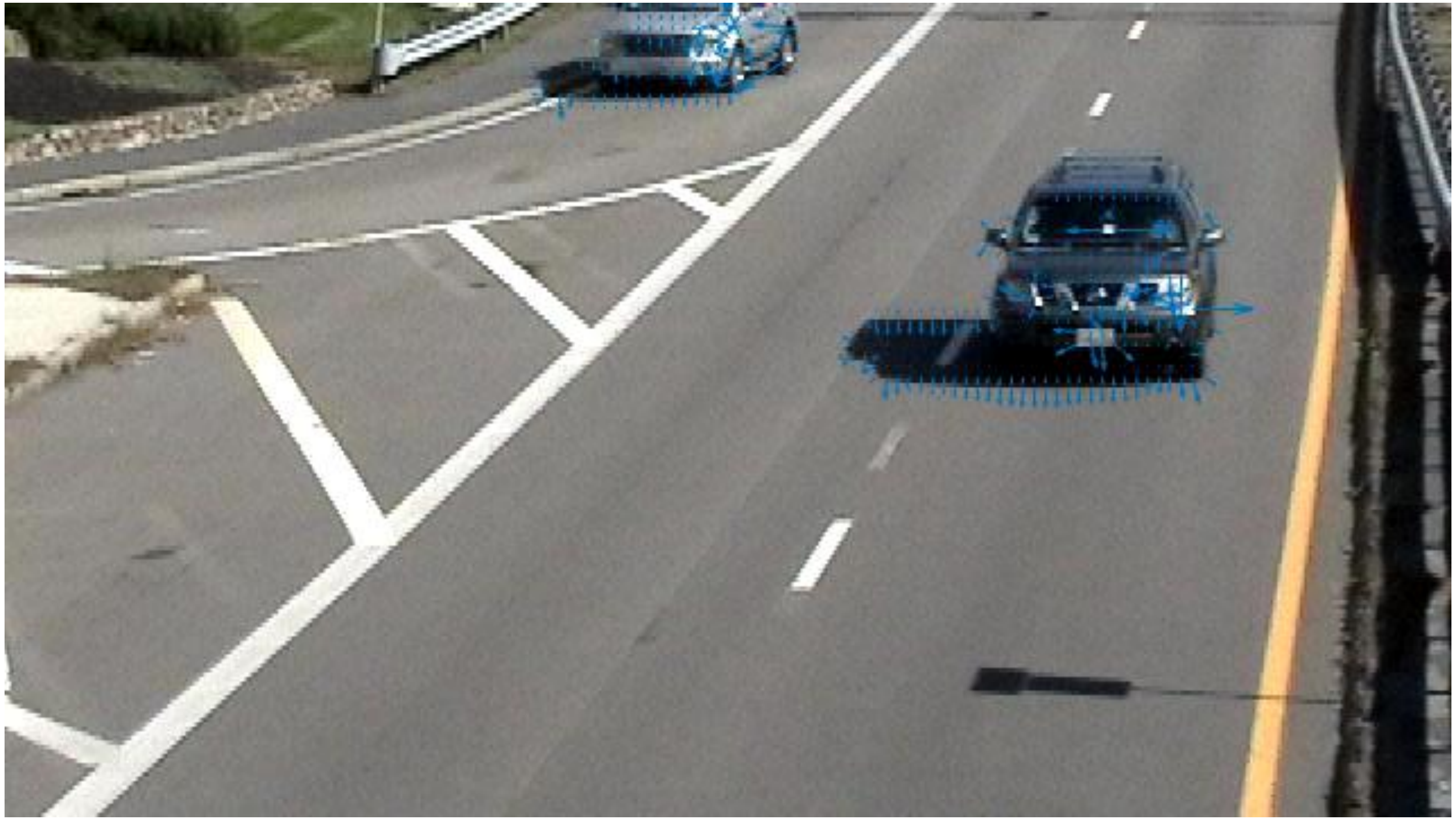
Megjegyzések Horn és Schunck módszeréhez

- Az egyes pixelekhez tartozó sebességvektorok kiszámításakor az egész képet figyelembe kell venni (az egyenletességi korlátozás miatt) → **globális technika**
- A fenti összefüggés minden pontra két egyenletet ad → hagyományos módszerrel már megoldható → De! időigényes
- Helyette iteratív megoldás: először egy durva becslés, utána lépésenként egyre finomítjuk (~30-100 iteráció)
- Fontos kérdés még a deriváltak kiszámításához használt módszer kiválasztása is

Horn és Schunck módszerére példa MATLAB környezetben

```
vidReader = VideoReader('visiontraffic.avi');
opticFlow = opticalFlowHS;
h = figure;
while hasFrame(vidReader)
    frameRGB = readFrame(vidReader);
    frameGray = rgb2gray(frameRGB);
    flow = estimateFlow(opticFlow,frameGray);
    imshow(frameRGB)
    hold on
    plot(flow,'DecimationFactor',[5 5],'ScaleFactor',60);
    hold off
    pause(10^-3)
end
```

Horn és Schunck MATLAB példa eredménye



Lucas és Kanade módszere

- Feltételezi, hogy a kép kisméretű szegmenseiben a sebesség állandó értékű → egy sebességvektorhoz több egyenlet tartozik
- Lokális technika: egy pont sebessége csak a lokális környezettől függ

Formálisan: Az optical flow korlátozásból kiindulva:

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

A vizsgált pixel és a négyzetes szomszédságában lévő képpontok sebessége ugyanaz. Tehát egy pont sebessége a környezetében lévő m x m méretű ablakot alapul véve (Ω tartomány), a t időpillanatban:

$$\begin{bmatrix} E_x(x_1, y_1) & E_y(x_1, y_1) \\ E_x(x_2, y_2) & E_y(x_2, y_2) \\ \vdots & \vdots \\ E_x(x_m, y_m) & E_y(x_m, y_m) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} E_t(x_1, y_1) \\ E_t(x_2, y_2) \\ \vdots \\ E_t(x_m, y_m) \end{bmatrix}$$

$$\mathbf{Av} = \mathbf{b}$$

Konstans lokális sebesség kényszere

Ha az [A|b] mátrixban pontosan 2 lineárisan független egyenlet van → egyértelmű megoldás

A redundancia miatt viszont valószínűleg több mint kettő a lineárisan független sorok száma → túlhatározott egyenletrendszer.

Ezért **minimalizálási célt** kell keresni és így kell megoldani:

$$|\mathbf{Av} - \mathbf{b}|^2 \rightarrow \min$$

$$\sum_{(x,y)\in\Omega} \left[E_x(x,y)u + E_y(x,y)v + E_t(x,y)\right]^2 \rightarrow \min$$

Konstans lokális sebesség kényszere

Ez egy legkisebb négyzet (LS) becslési probléma → létezik „szabványos” megoldás:

$$\mathbf{A}^T\mathbf{A}\mathbf{v} = \mathbf{A}^T\mathbf{b}$$

$$\mathbf{v} = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T\mathbf{b}$$

Megjegyzések:

A megoldhatósághoz az $\boldsymbol{A}^T\boldsymbol{A}$ mátrixnak invertálhatónak kell lennie

- Gyengén textúrázott esetben ez a mátrix közel szinguláris lesz → nincs megoldás
- Az objektumhatárok mentén továbbra is probléma van

A megoldás általánosítása/javítása

- A vizsgált tartományon belüli sebességek ugyanakkorák → túl „naiv” feltételezés
- Ahogy haladunk a magas felbontás felé, a sebességek egyre jobban változhatnak

Ezért érdemes az egyes pixelekhez súlyokat rendelni (általában Gauss eloszlás szerint)

Ezzel a korábbi egyenletek így módosulnak:

$$\mathbf{WAv} = \mathbf{Wb}$$

$$\sum_{\mathbf{x}\in\Omega} \mathbf{W}^2(\mathbf{x})\left[\nabla E(\mathbf{x})\cdot\mathbf{v} + E_t(\mathbf{x})\right]^2$$

$$\mathbf{A}^T\mathbf{W}^2\mathbf{Av} = \mathbf{A}^T\mathbf{W}^2\mathbf{b}$$

$$\mathbf{v} = \left(\mathbf{A}^T\mathbf{W}^2\mathbf{A}\right)^{-1}\mathbf{A}^T\mathbf{W}^2\mathbf{b}$$

Megjegyzések Lucas és Kanade módszeréhez

Hátrány:

- Csupán kisméretű elmozdulásokat tud detektálni

Megoldás: iteratív eljárás

1. Elsőként az eredeti kép alacsony felbontású változatán hajtjuk végre az algoritmust (nagy mozgás is csak néhány pixel elmozdulást jelent) (A képi piramisok alkalmazását már korábban is láttuk!)

2. Ezután egyre finomítjuk a vektormezőt

Korrelációs technikák

Eddig differenciális technikákat használtunk → numerikus differenciálás

Ez problémát jelenthet, ha:

- Alacsony a jel-zaj viszony (zajos a kép)
- Alacsony a képkockák száma
- A képnyerési folyamatban jelentkező torzításoknál

Ekkor jönnek számításba a korrelációs technikák

Illeszkedések leírása

A kép *n* x *n* méretű régióit vizsgáljuk az egyes pixelek helyett

Összevetjük a *t*. kép egy régióját a *t*+1. kép régióival → Hozzárendelünk minden illesztéshez egy arányszámot. Két gyakori technika:

Keresztkorreláció (Cross-Correlation)

$$CC_{1,2}(\mathbf{x},\mathbf{d}) = \sum_{j=-n}^{+n} \sum_{i=-n}^{+n} W(i,j)\left[E_1(\mathbf{x}+(i,j))E_2(\mathbf{x}+(i,j)+\mathbf{d})\right]$$

Eltérések négyzetösszege (Sum of Squared Differences)

$$SSD_{1,2}(\mathbf{x},\mathbf{d}) = \sum_{j=-n}^{+n} \sum_{i=-n}^{+n} W(i,j)\left[E_1(\mathbf{x}+(i,j)) - E_2(\mathbf{x}+(i,j)+\mathbf{d})\right]^2$$

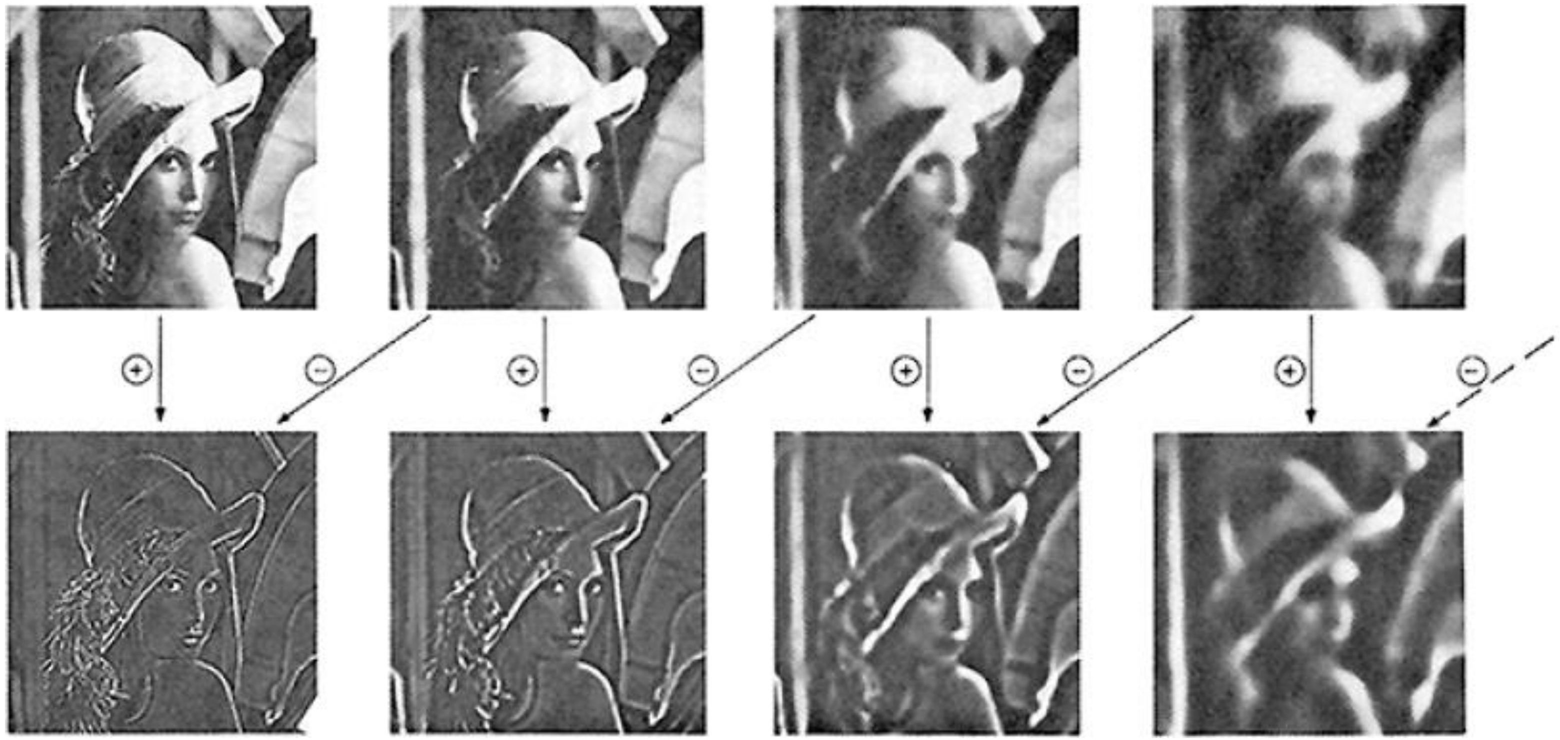
Végül (attól függően, hogy világos vagy sötét régiót követünk) a legnagyobbat / legkisebbet kiválasztva kapjuk meg az elmozdulást.

Megjegyzések a korrelációs módszerekhez

A szomszédos pixelek erős összefüggésben állnak egymással

Az optical flow számítások hatékonyabbá tehetők, ha csökkentjük ezt a kapcsolatot

Példa: Laplace piramis



Spektrumképre épülő módszerek

- Néhány esetben érdemes frekvenciatérben végrehajtani az optical flow számításokat.

  Pl. véletlen pontmintázatok mozgása → nincsen elég információ a differenciális vagy a korrelációs módszerekhez

- A korábbiakhoz hasonlóan fel lehet írni az állandósági összefüggéseket

Két csoport:

- Energiára épülő technikák → amplitúdót használják
- Fázisra épülő technikák → fázist veszik figyelembe

Mozgás alapú szegmentálás

Szegmentálás az optical flow mezőn

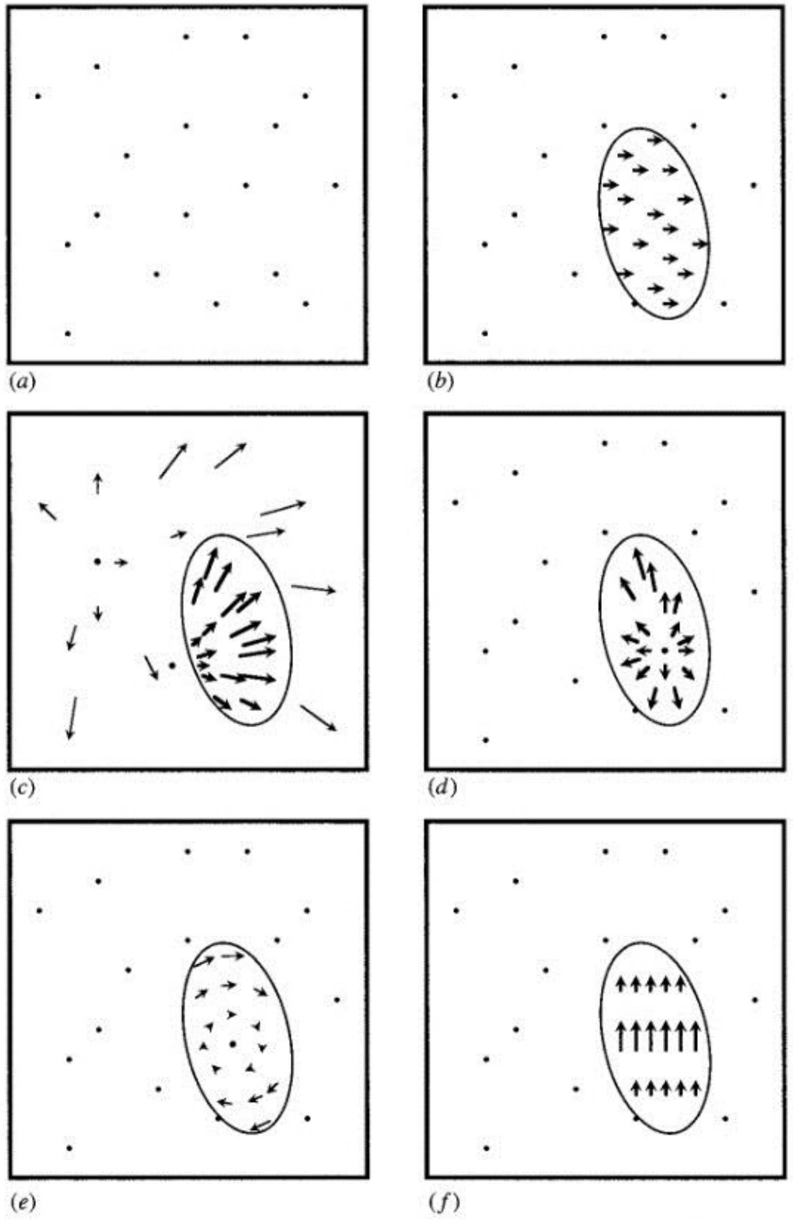
Mi legyen a hasonlósági kritérium?

Optical flow mintapéldák

Az objektum áll

A kamera mozog, az objektum más irányban szintén

Az objektum az optikai tengely körül forog



Az objektum halad (jobbra)

A kamera áll, az objektum közelít

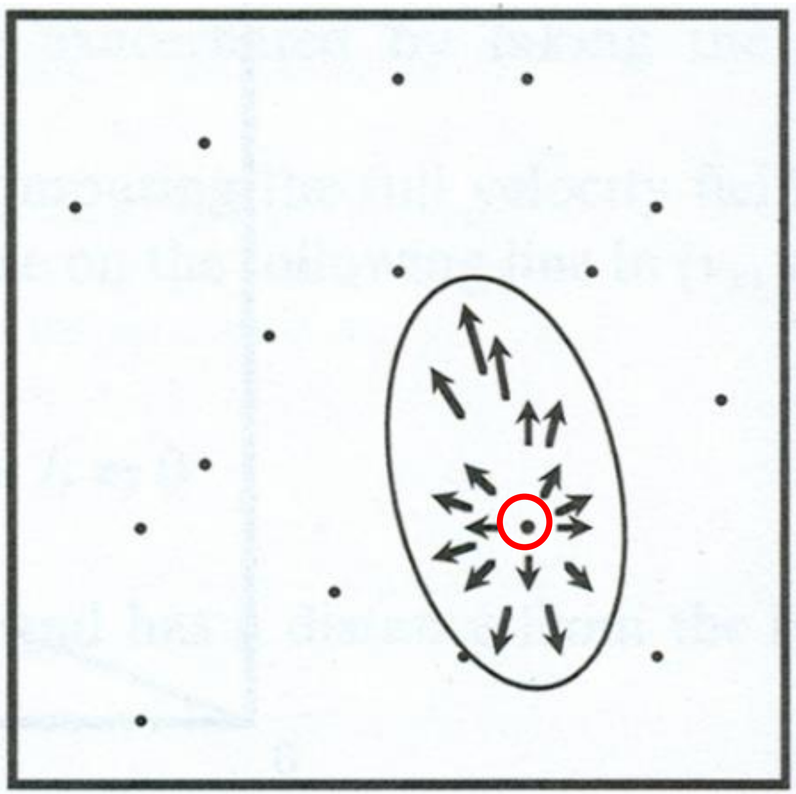
Az objektum az optikai tengelyre merőleges tengely körül forog
ld. korábban a gömb

Optical flow alkalmazás - Tágulási fókusz

Tágulási fókusz (focus of expansion – FoE) az a pont, ami felé közeledünk.

Felhasználható pl.:

- Akadálykerülés
- Ütközés-előrejelzés



Köszönöm a figyelmet!