

# JOHDATUS TEKOÄLYYN

TEEMU ROOS



HELSINGIN YLIOPISTO



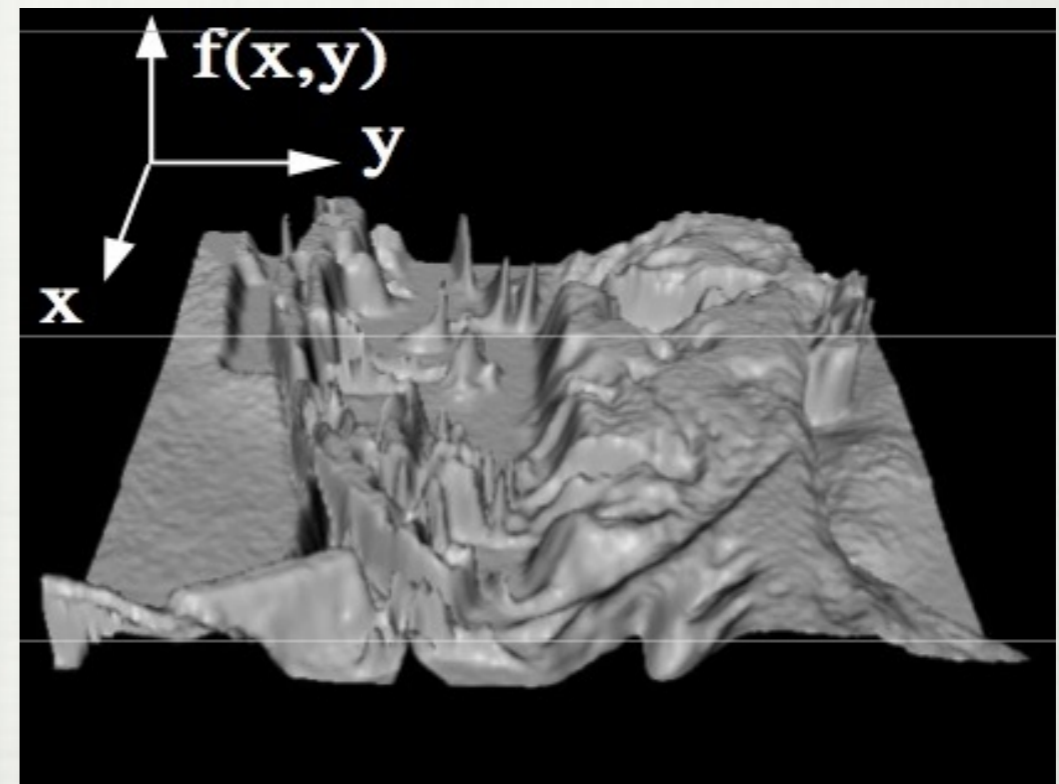


TERMINATOR  
SALVATION



# SIGNAALINKÄSITTELY

- \* KUVA VOIDAAN TULKITA KOORDINAATTIEN  $(x,y)$  FUNKTIONA.



LÄHDE: S. SEITZ

- \* VÄRIKUVA KOOSTUU KOLMESTA KOMPONENTISTA (R,G,B).
- \* ÄÄNI VASTAAVASTI MUUTTUJAN  $t$  (AIKA) FUNKTIONA (TAI MUUTTUJAN  $f$  (TAAJUUS)  $\Rightarrow$  "FREQUENCY DOMAIN")

# SIGNAALINKÄSITTELY

- \* KUVA VOIDAAN TULKITA KOORDINAATTIEN (X,Y) FUNKTIONA.

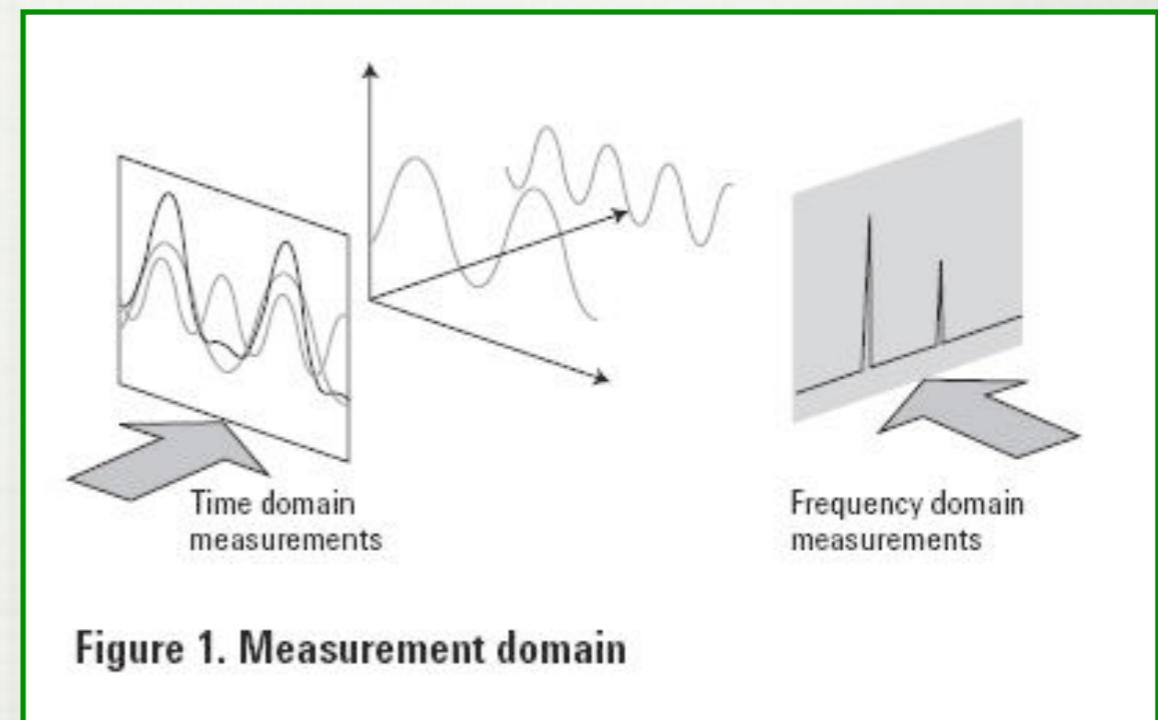
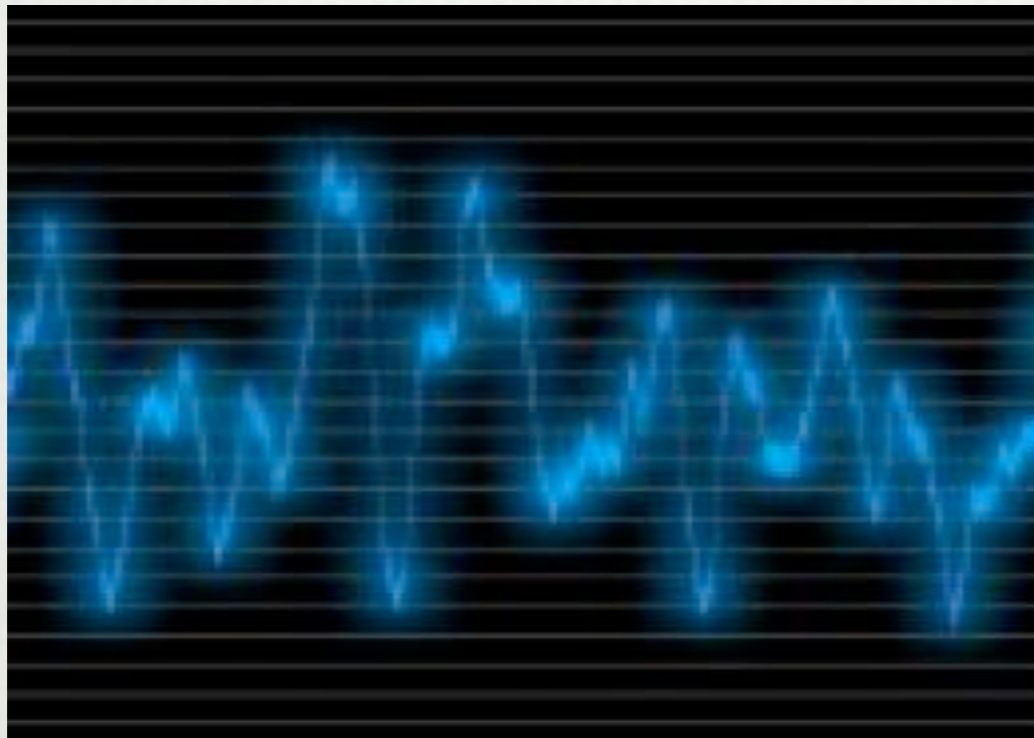


Figure 1. Measurement domain

- \* VÄRIKUVA KOOSTUU KOLMESTA KOMPONENTISTA (R,G,B).
- \* ÄÄNI VASTAAVASTI MUUTTUJAN T (AIKA) FUNKTIONA (TAI MUUTTUJAN F (TAAJUUS) => "FREQUENCY DOMAIN")



# SIGNAALINKÄSITTELY

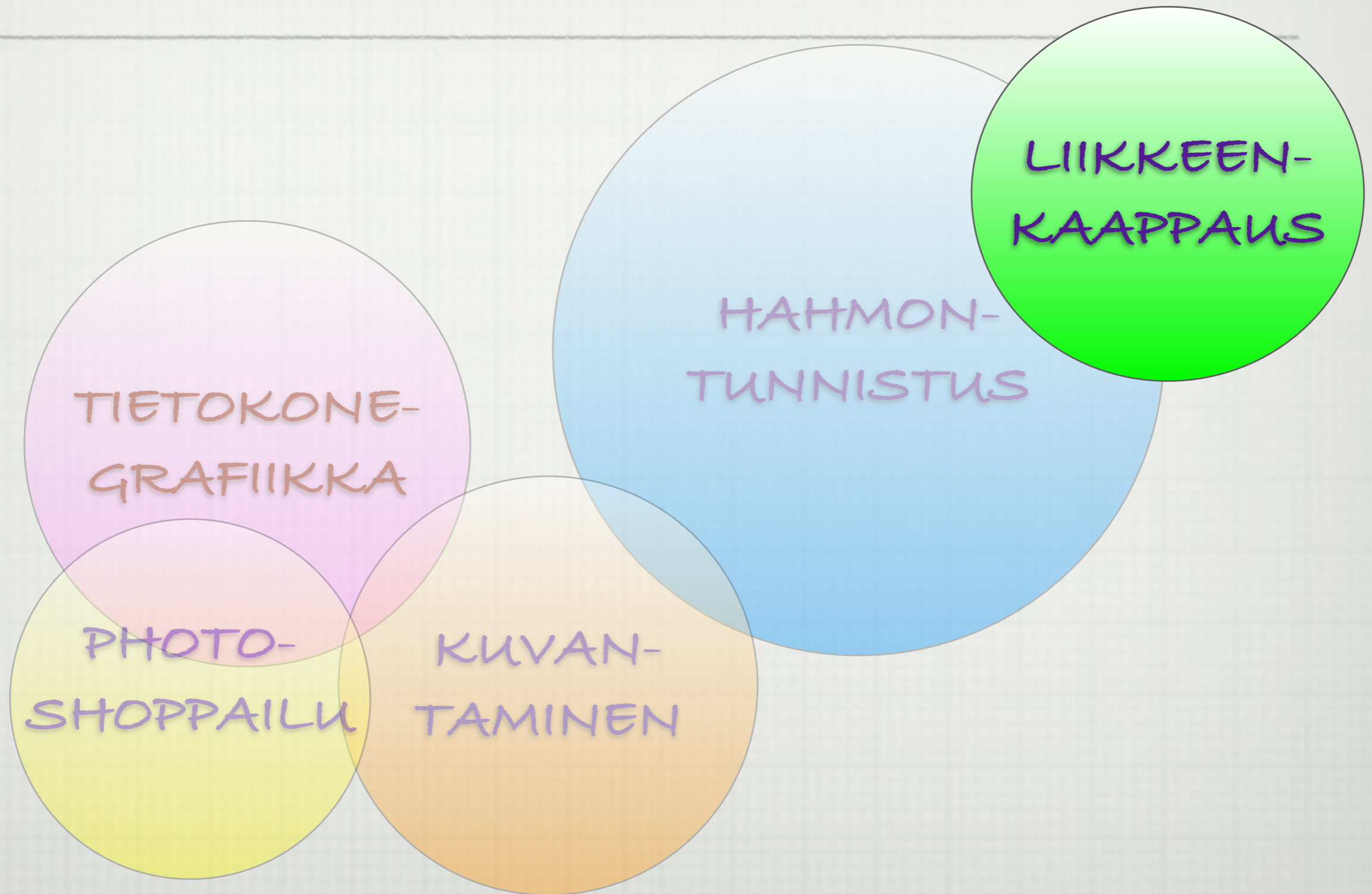
---

- \* KUVAN TAI ÄÄNITIEDOSTON **KOKO** ON USEIN SUURI:  
ESIM. 1000 X 1000 PIKSELIÄ = 1 000 000 PIKSELIÄ
  - KUVA: **RESOLUUTIO** [DPI]
  - ÄÄNI: **NÄYTTEENOTTOTAAJUUS** [HZ]
- \* **KOHINAA** ESIINTYY AINA LUONNOLLISESSA SIGNAALISSA
- \* SIGNAALI RIIPPUU MYÖS **OLOSUHTEISTA**:
  - KUVAKULMA, VALAISTUS, ETÄISYYS, ...
  - KAIKU, MIKROFONI, TAUSTAHÄLY, ...
- \* ⇒ TUNNISTAMINEN TAI LUOKITTELU VAIKEAA



# SIGNAALINKÄSITTELY

---



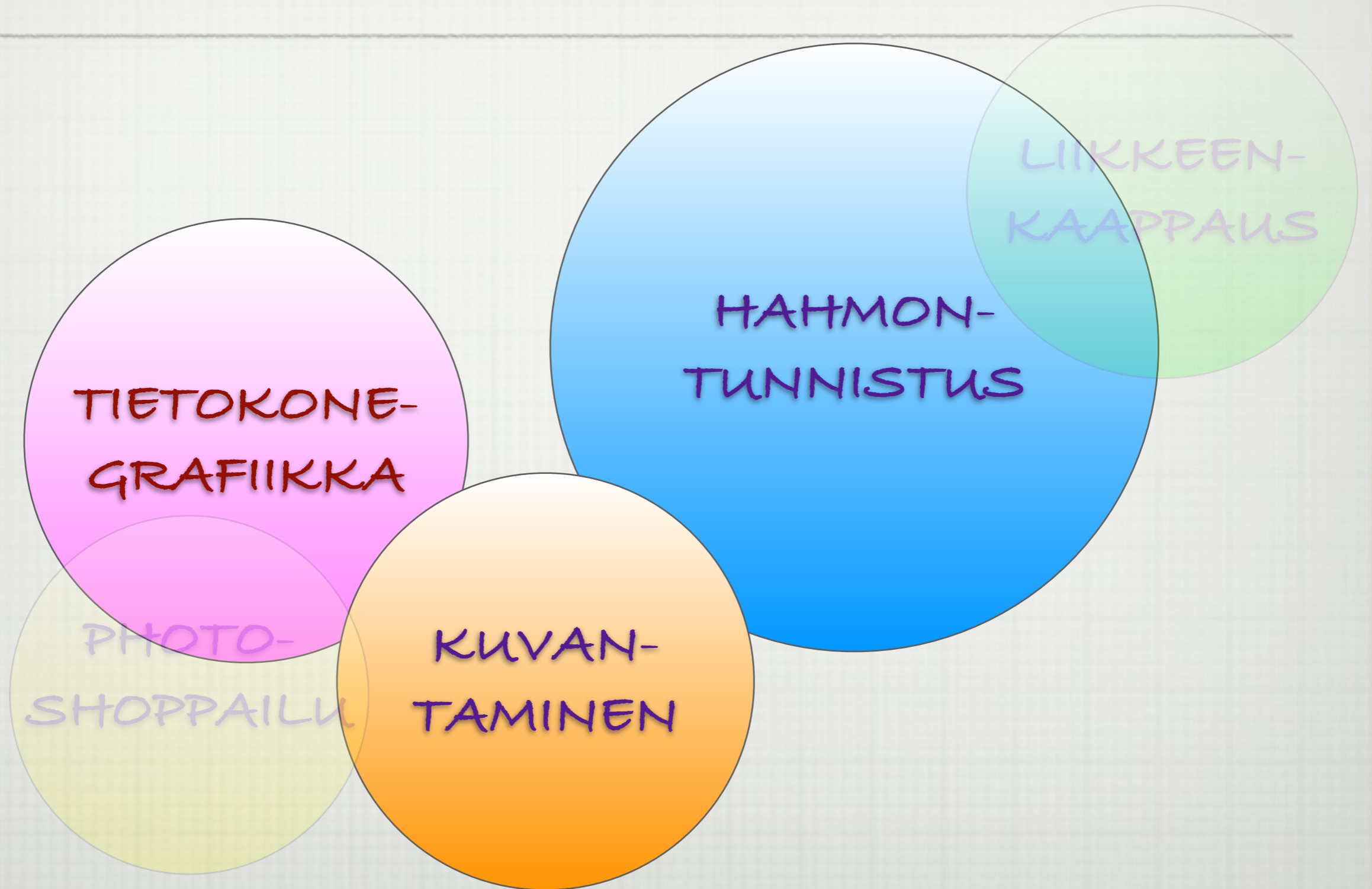


# MOTION CAPTURE



# SIGNAALINKÄSITTELY

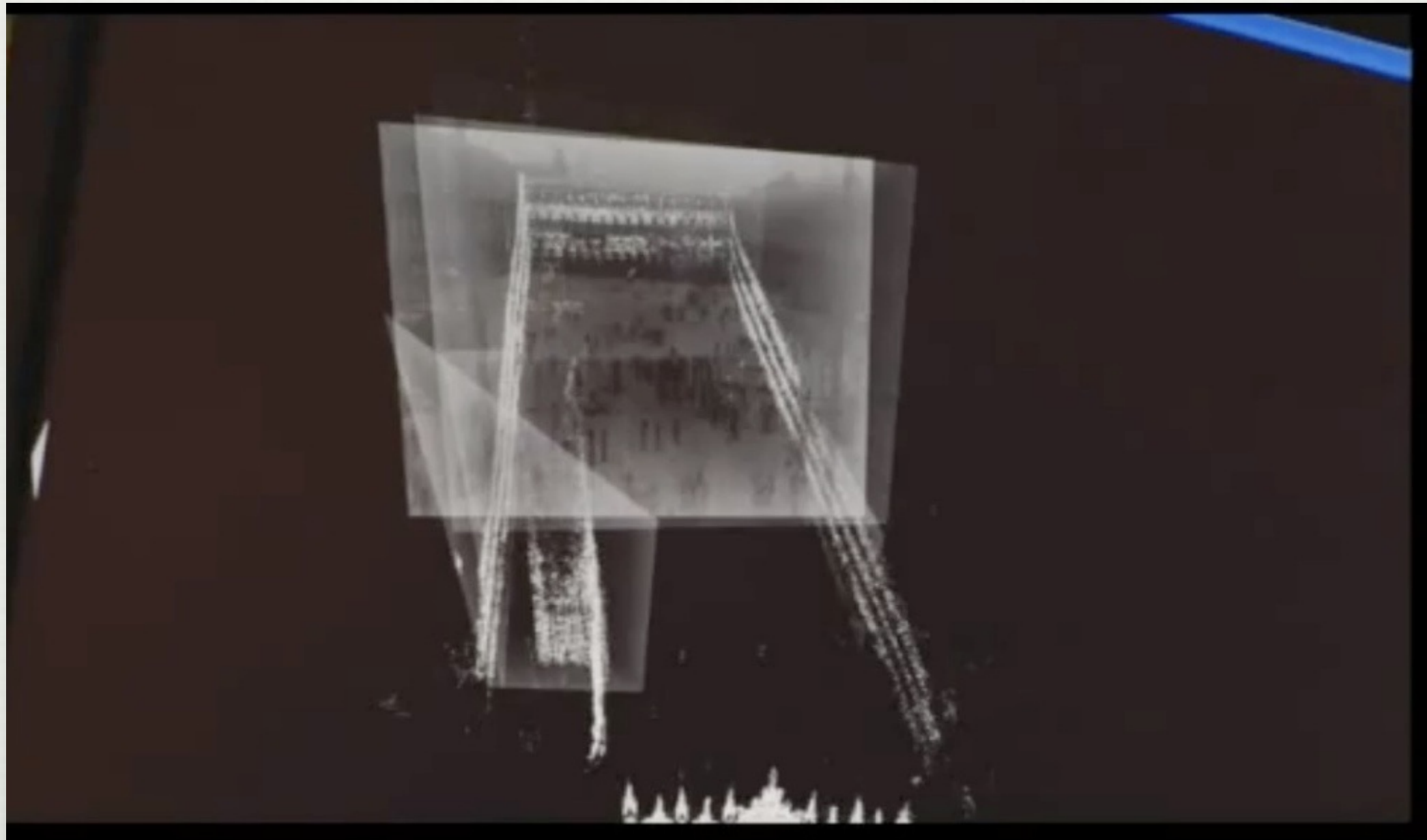
---





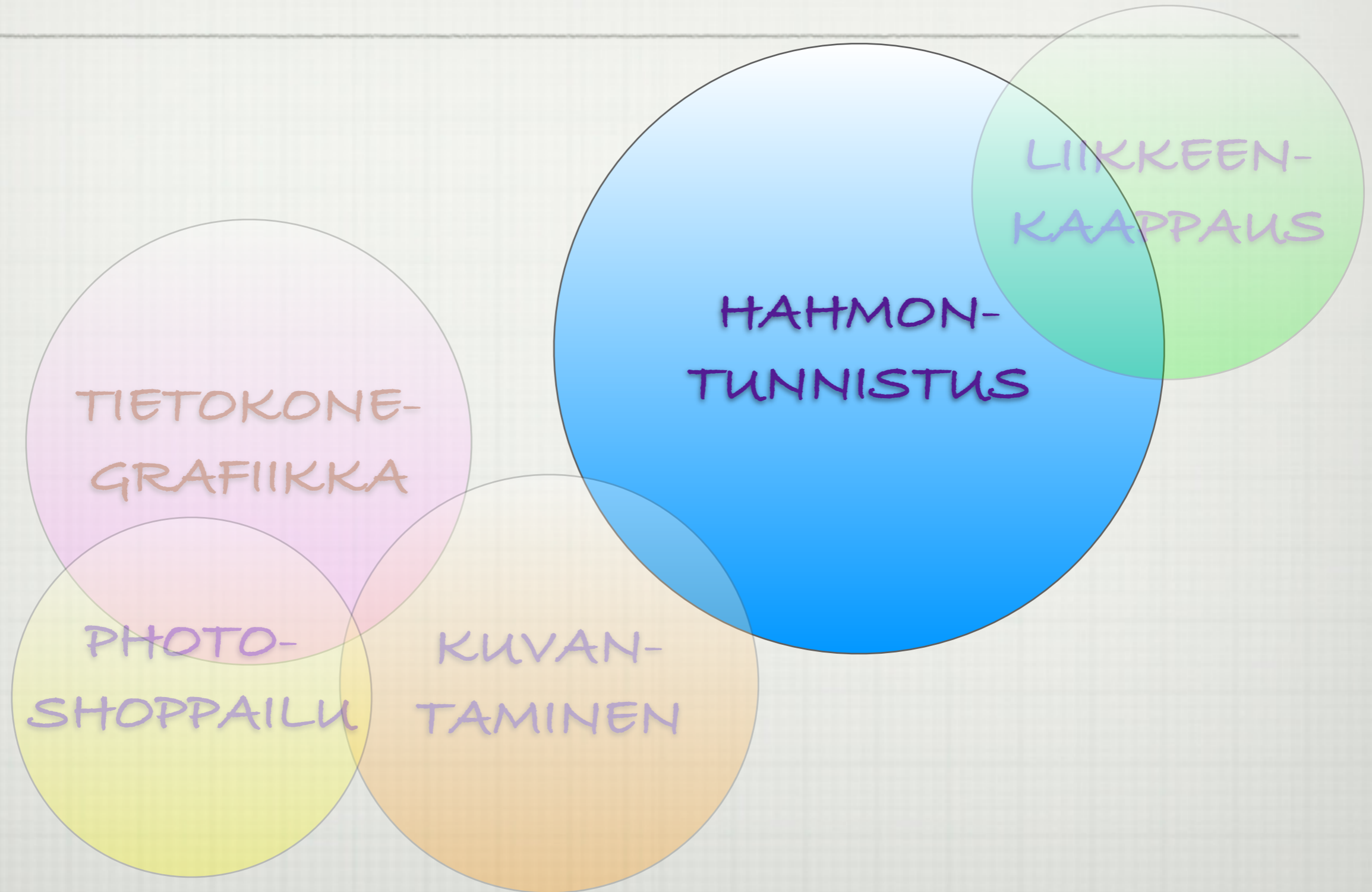
# MICROSOFT PHOTOSYNTH

---



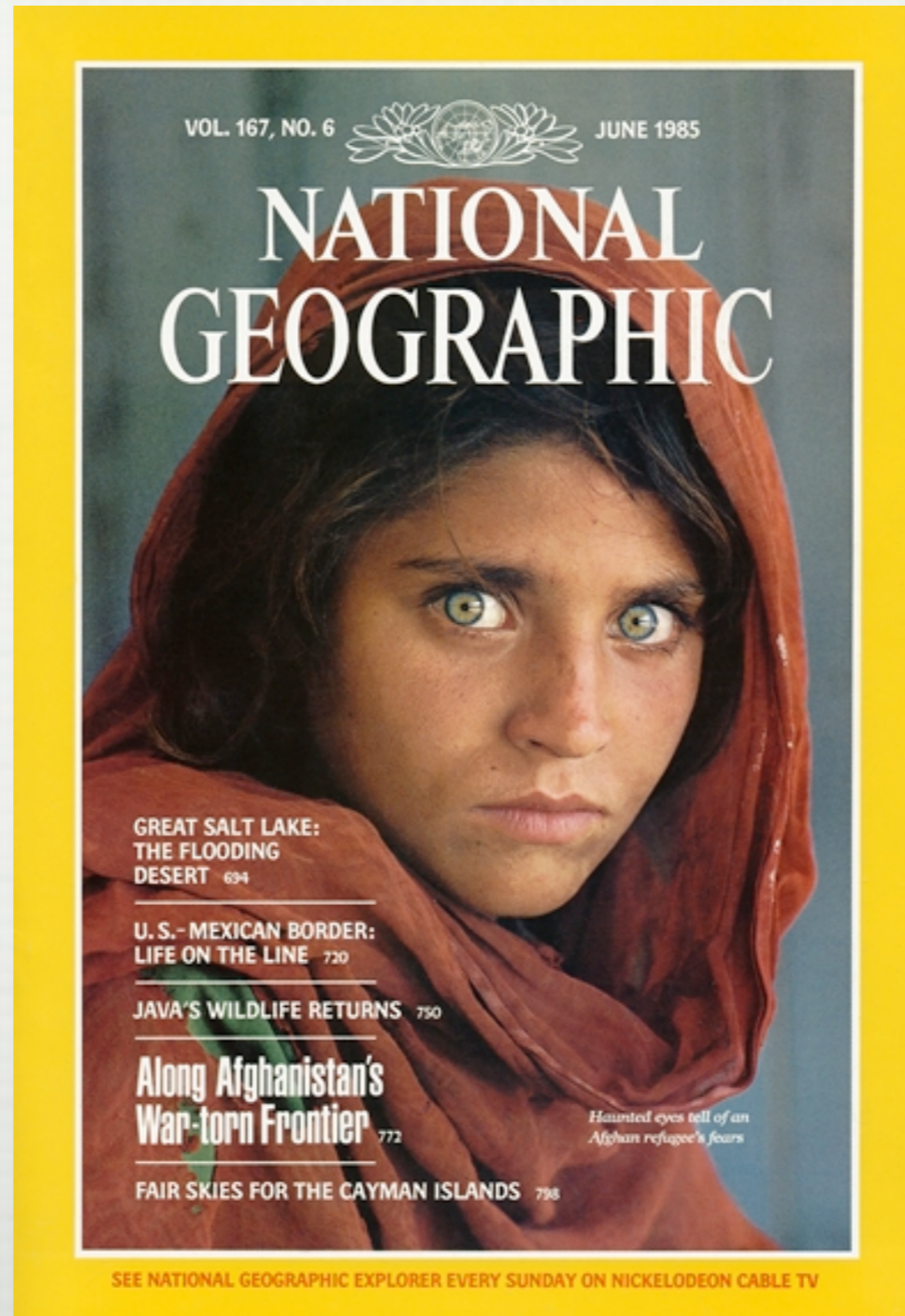
# SIGNAALINKÄSITTELY

---





# “AFGHAN GIRL”





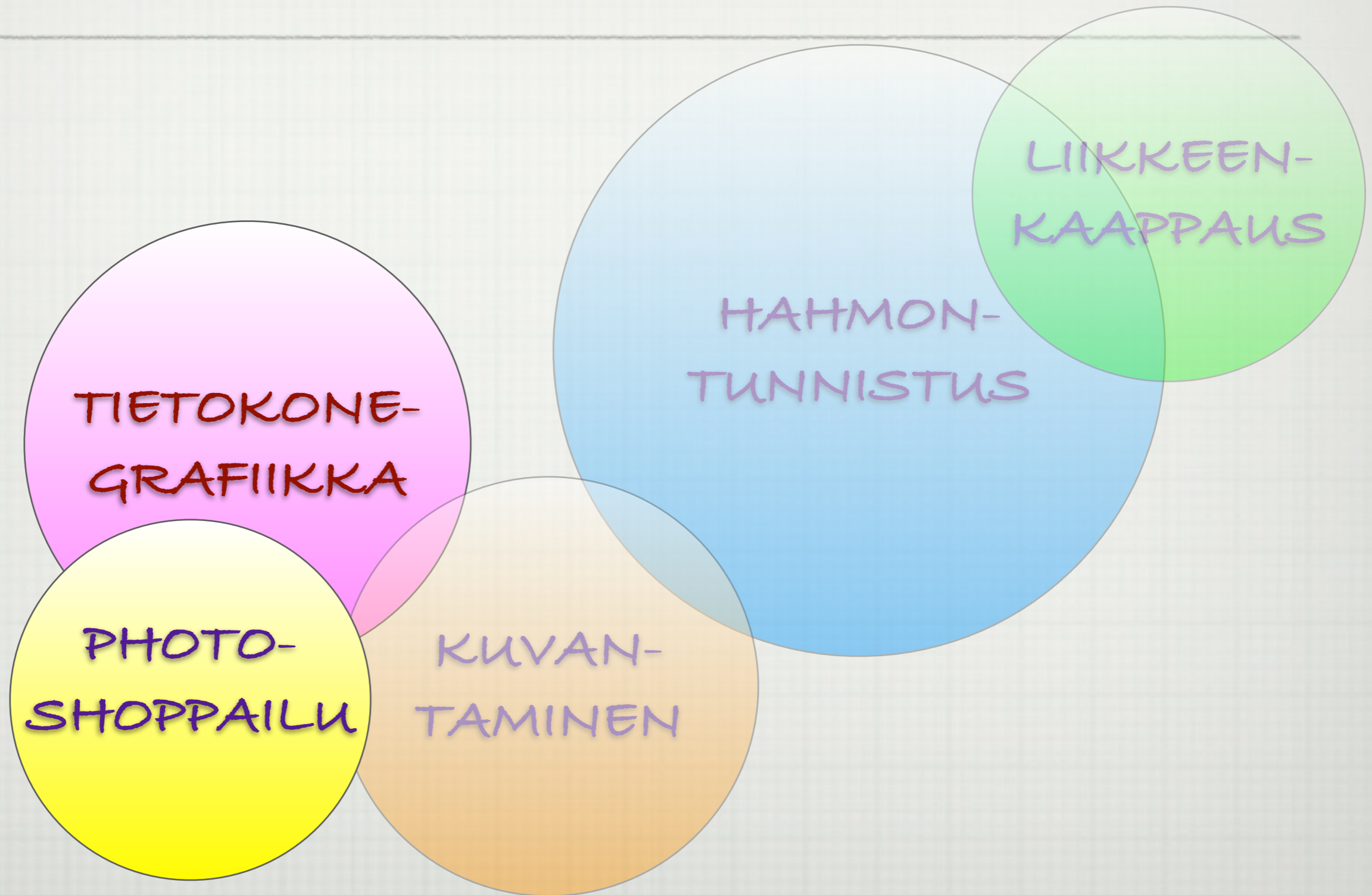
# “AFGHAN GIRL”





# SIGNAALINKÄSITTELY

---

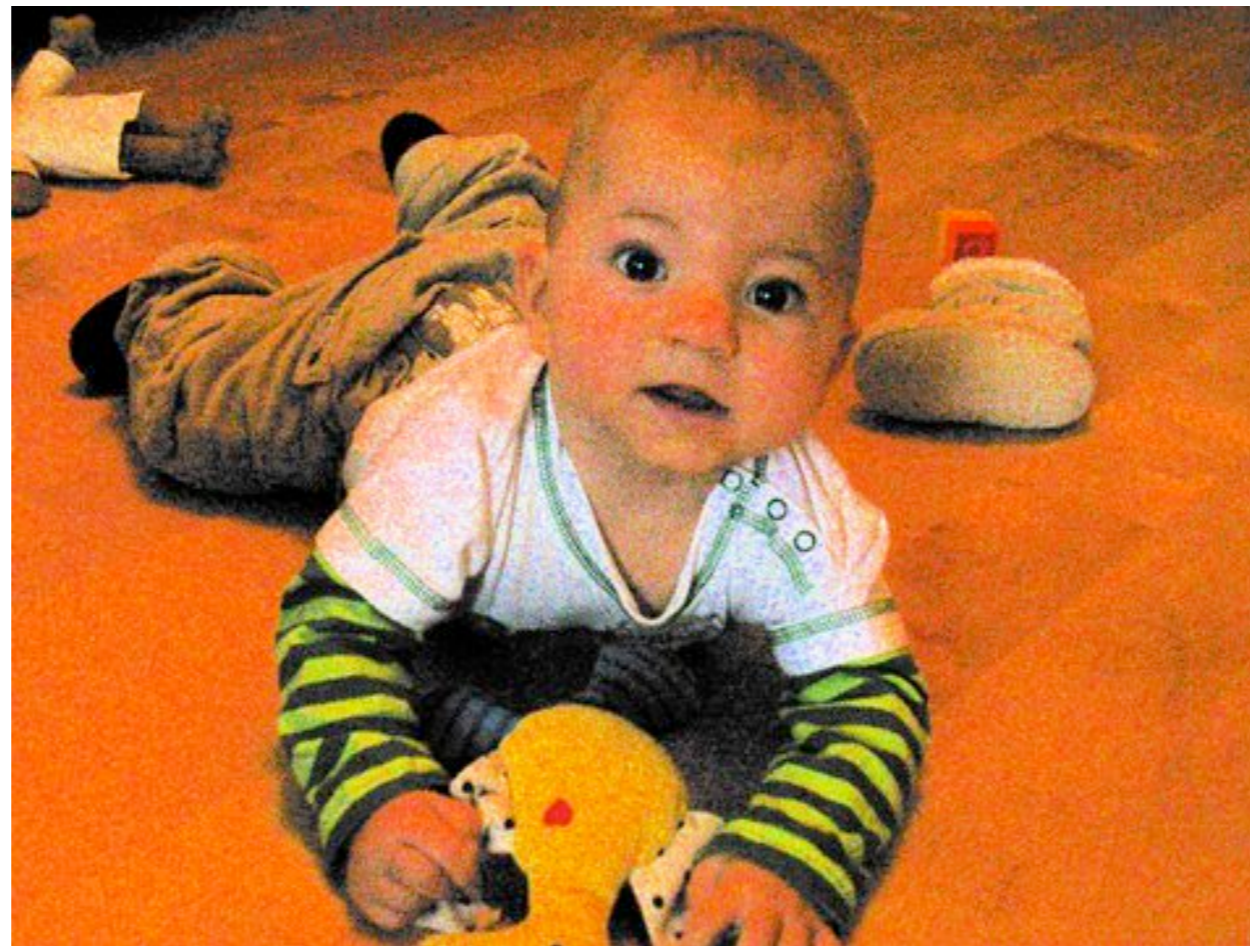


# KOHINANPOISTO

Occam's Razor  
MDL Principle  
Universal Source Coding  
MDL Principle (contd.)

Modern MDL  
Histogram Density Estimation  
Clustering  
Linear Regression  
Wavelet Denoising

## Example: Denoising

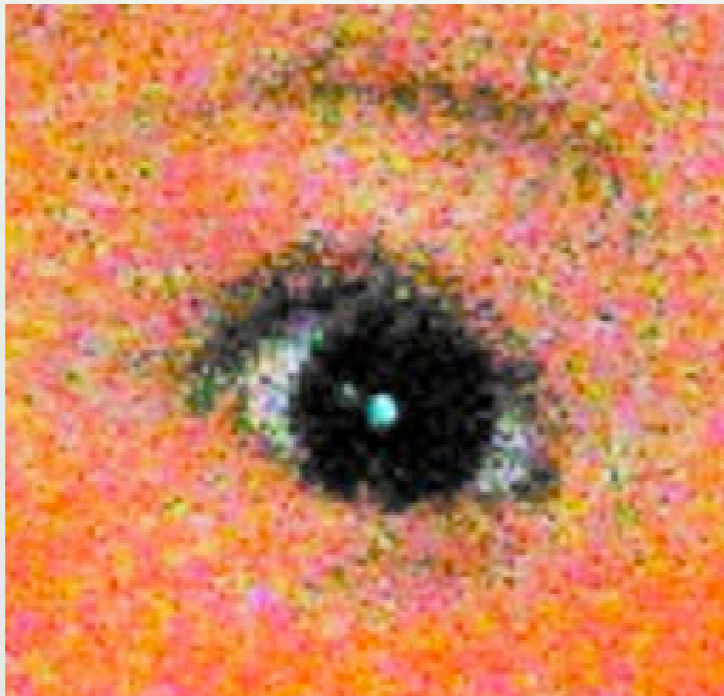




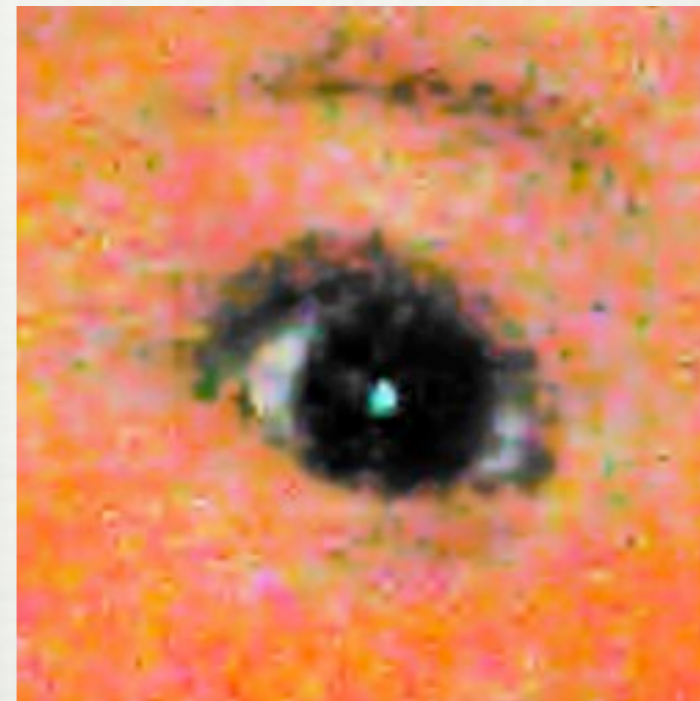
# KOHINANPOISTO

---

ALKUPERÄINEN

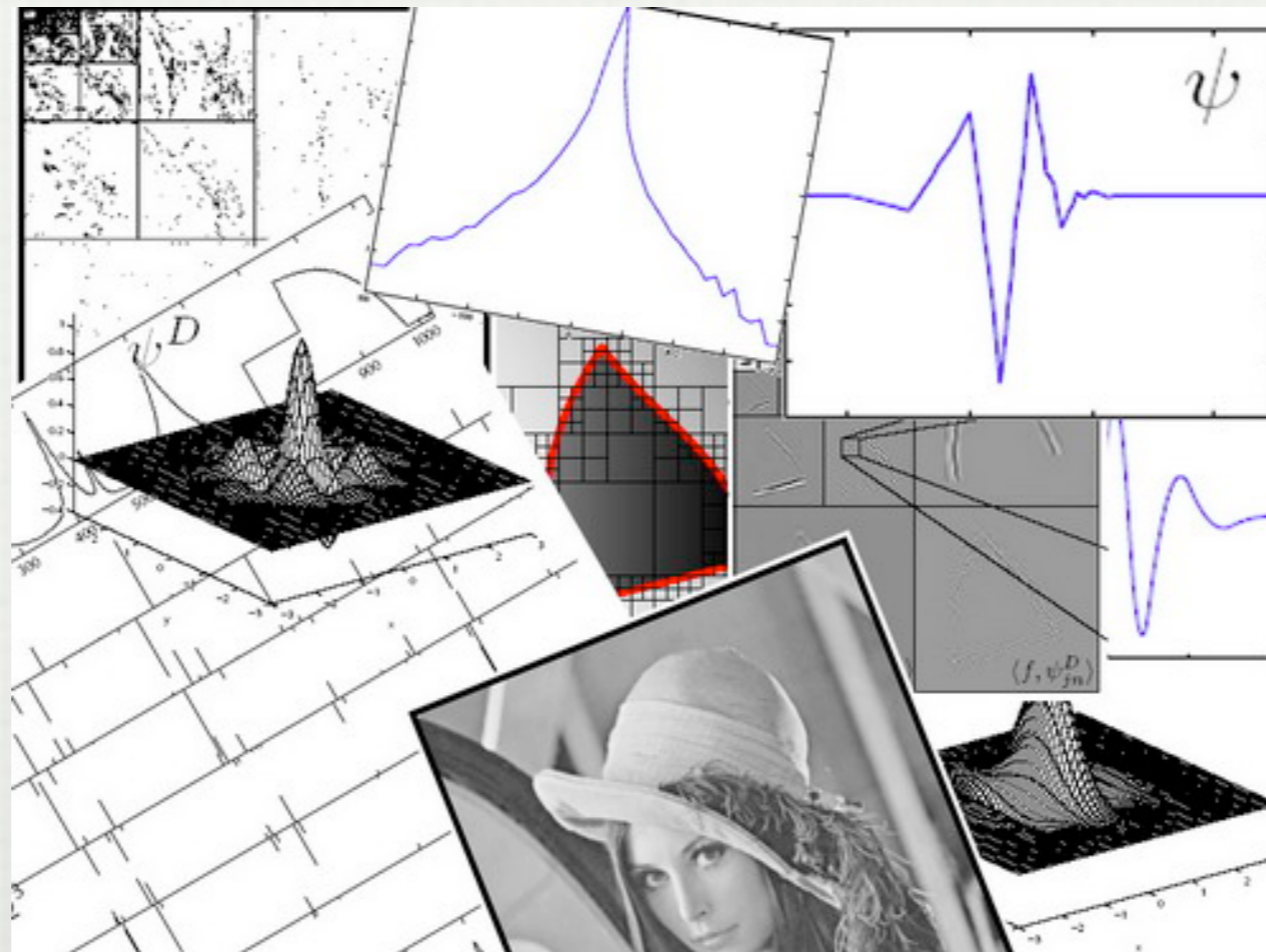


KÄSITELTY



# KOHINANPOISTO

- \* SIGNAALI (ÄÄNI TAI KUVA) ESITETÄÄN NIIN SANOTUSSA AALLOKEMUODOSSA (WAVELET)



- \* HIUKAN KUIN TAAJUUS-ESITYS ÄÄNELLÄ

- \* KÄSITELLÄÄN SIGNAALIA VEKTORINA



# KOHINANPOISTO

- \* AALLOKEMUODOSSA SIGNAALIN OLENNAINEN OSA KESKITTYNYT VAIN HARVOIHIN, ISOIHIN ELEMENTTEIHIN  
[0,0,0, 45, -12, 0,0,0,0,-2, 0,0,-120, 0,0,0,...]
- \* KOHINA JAKAUTUU TASAISESTI KAIKILLE ELEMENTEILLE  
[ε,ε,ε, 45±ε, -12±ε, ε,ε,ε,ε, -2±ε, ε,ε,ε, -120±ε, ε,ε,ε,...]
- \* MENETELMÄ: ASETA NOLLAA LÄHELLÄ OLEVAT ELEMENTIT NOLLIKSI  
[0,0,0, 45±ε, -12±ε, 0,0,0,0, 0,0,0, -120±ε, 0,0,0,...]
- \* TULOS:
  - SUURIN OSA KOHINASTA HÄVIÄÄ
  - VAIN PIENI OSA SIGNAALISTA HÄVIÄÄ

# KOHINANPOISTO

---

kohinanpoisto(x,t):

n ← length(x)

c ← dwt(x) // discrete wavelet transform

for i = 1,...,n:

    if |c[i]| < t: c[i] = 0

end-for

x ← idwt(c) // inverse wavelet transform

return x



# KOHINANPOISTO

---

kohinanpoisto(x,t):

AALLOKEMUUNNOS

n ← length(x)

c ← dwt(x) // discrete wavelet transform

for i = 1,...,n:

    if |c[i]| < t: c[i] = 0

end-for

x ← idwt(c) // inverse wavelet transform

return x

# KOHINANPOISTO

---

kohinanpoisto(x,t):

n ← length(x)

c ← dwt(x) // discrete wavelet transform

for i = 1,...,n:

if  $|c[i]| < t$ : c[i] = 0

end-for

x ← idwt(c) // inverse wavelet transform

return x

KYNNYSARVO t





# KOHINANPOISTO

---

kohinanpoisto(x,t):

n ← length(x)

c ← dwt(x) // discrete wavelet transform

for i = 1,...,n:

if |c[i]| < t: c[i] = 0

end-for

x ← idwt(c) // inverse wavelet transform

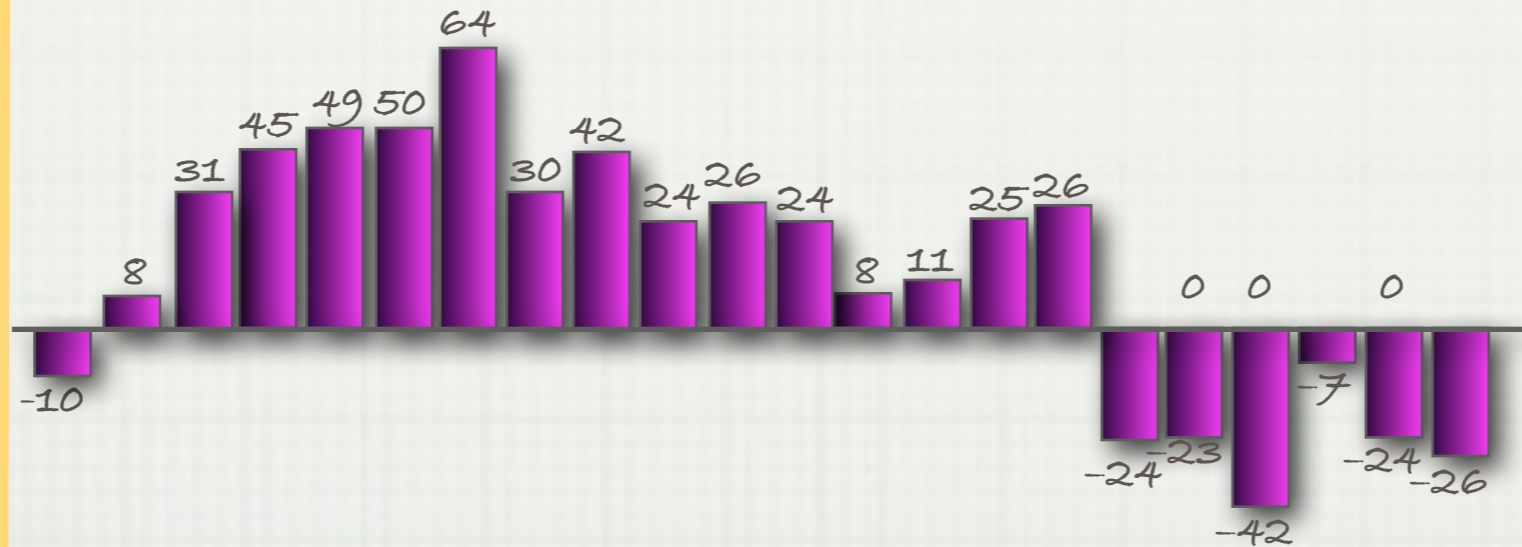
return x

KÄÄNTEISMUUNNOS



# KOHINANPOISTO

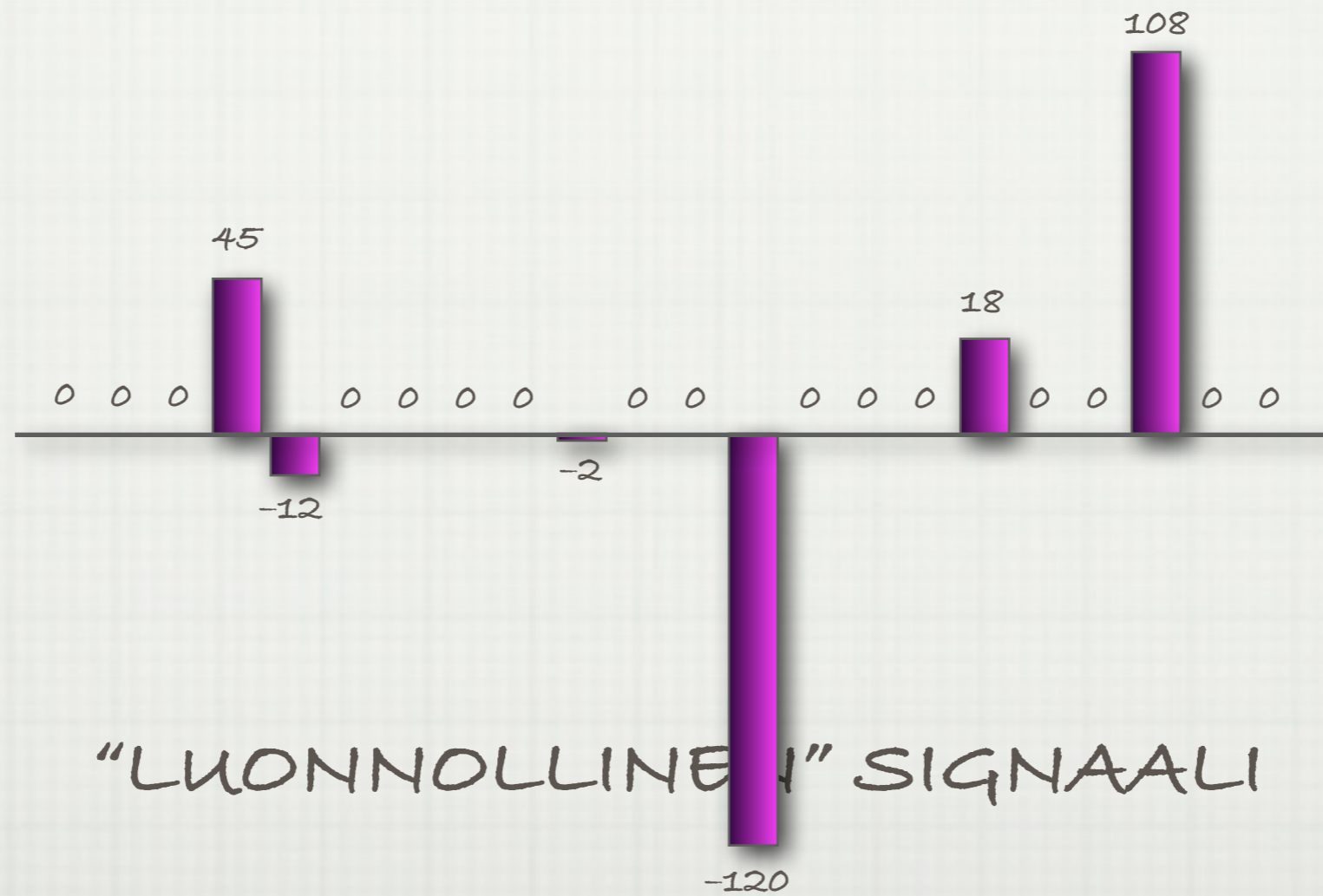
AALLOKEMUUNNOS



"LUONNOLLINEN" SIGNAALI



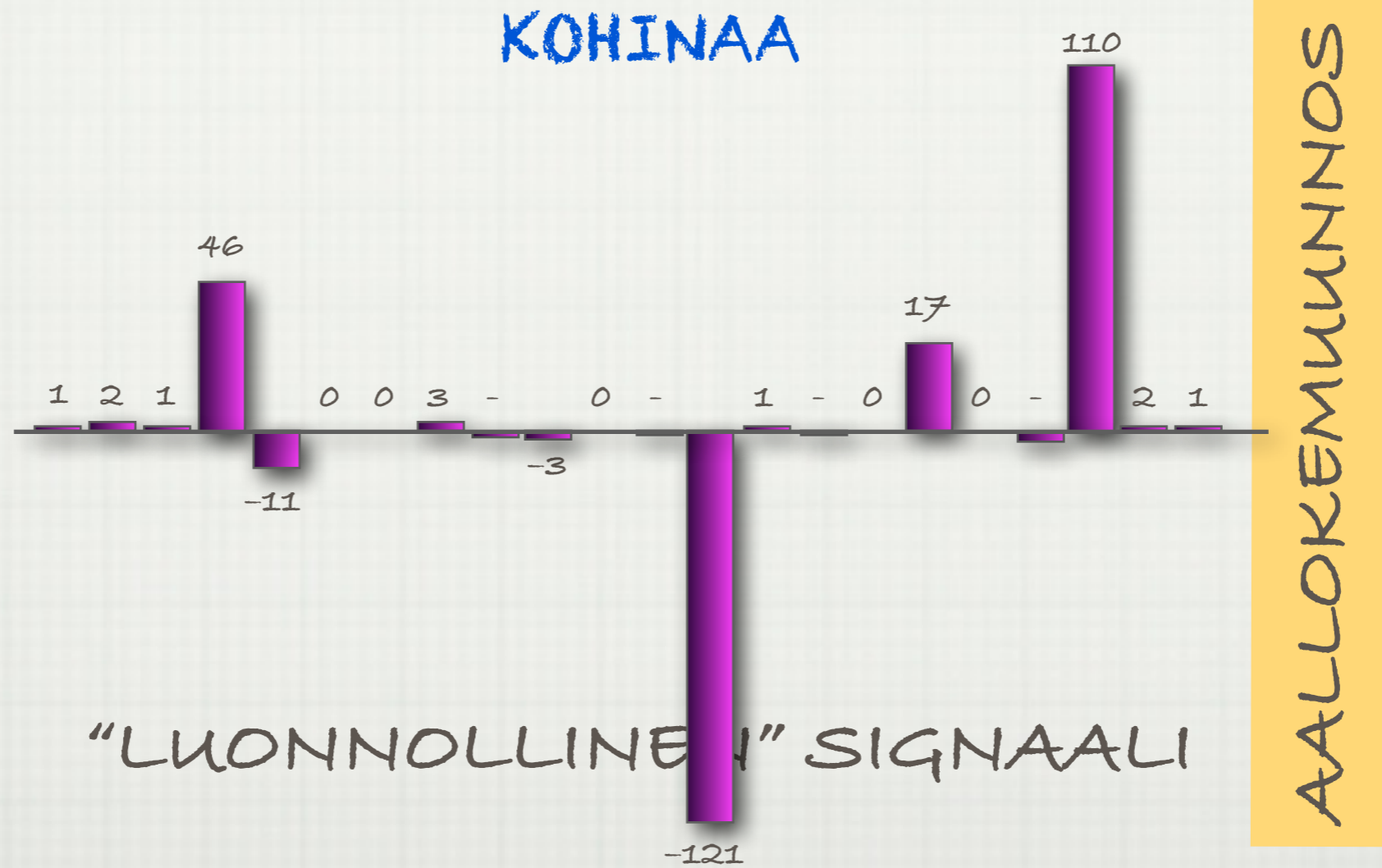
# KOHINANPOISTO



"LUONNOLLINEN" SIGNAALI

AALLOKEMUUNNOS

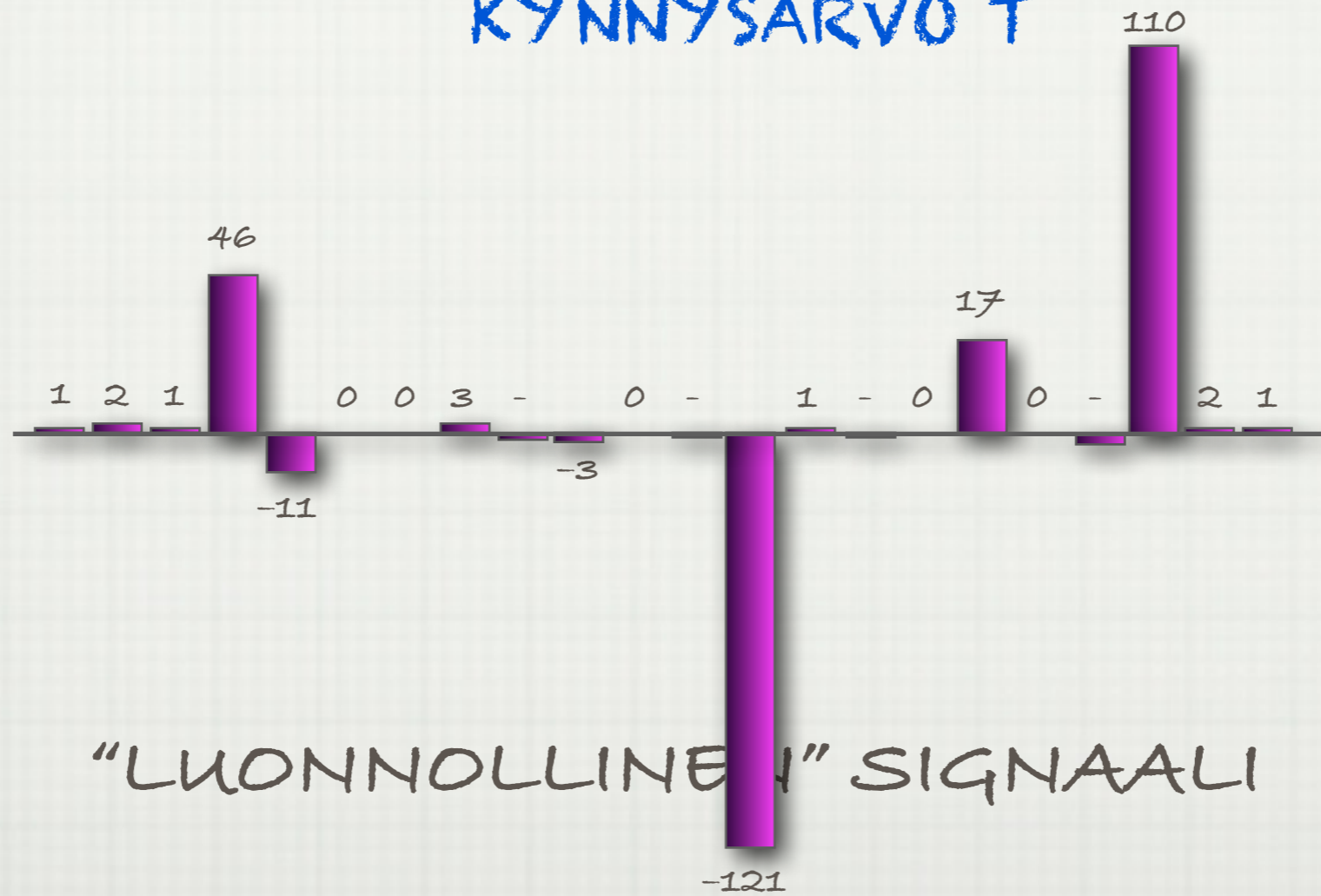
# KOHINANPOISTO





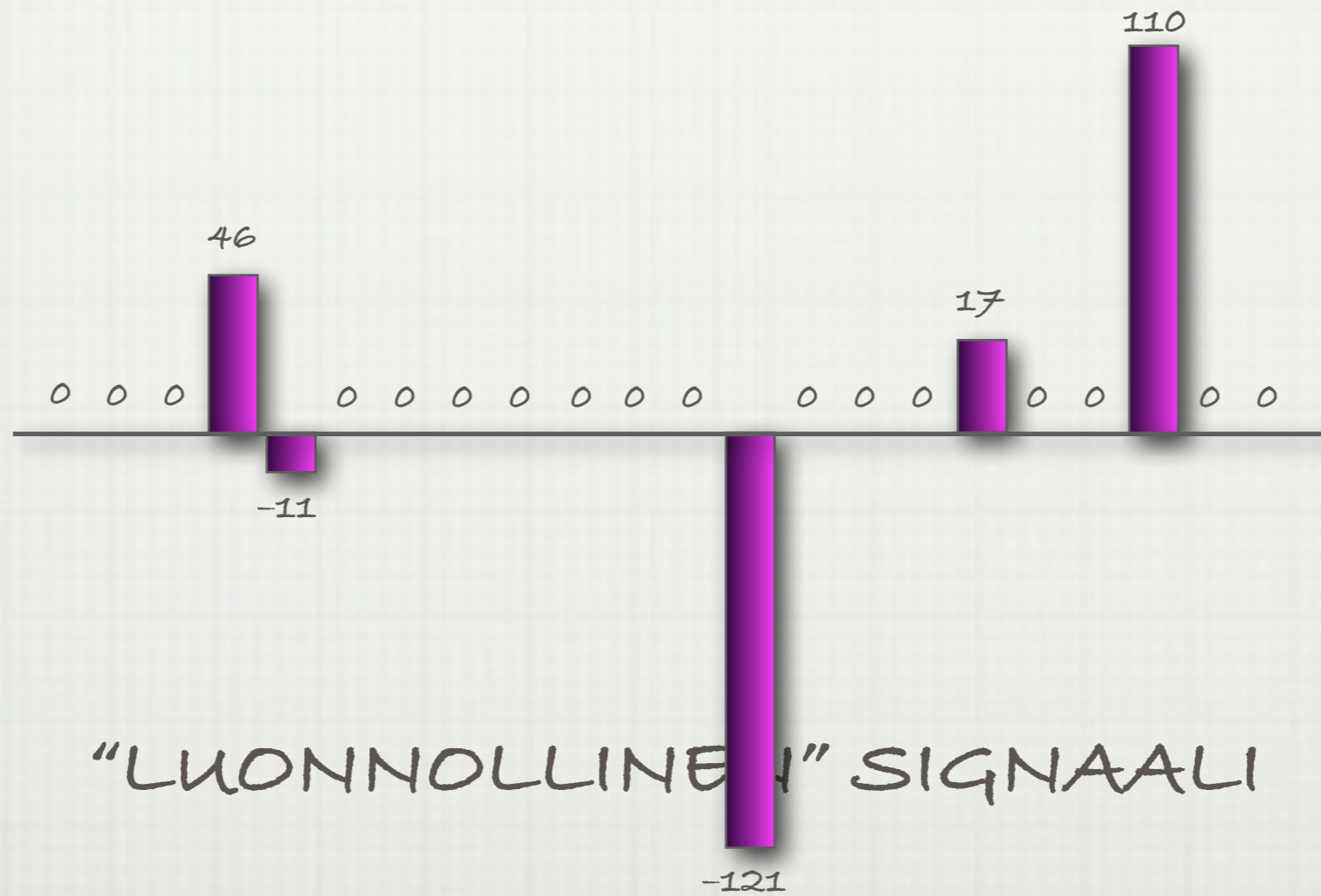
# KOHINANPOISTO

KYNNYSARVOT



"LUONNOLLINEN" SIGNAALI

# KOHINANPOISTO



"LUONNOLLINEN" SIGNAALI

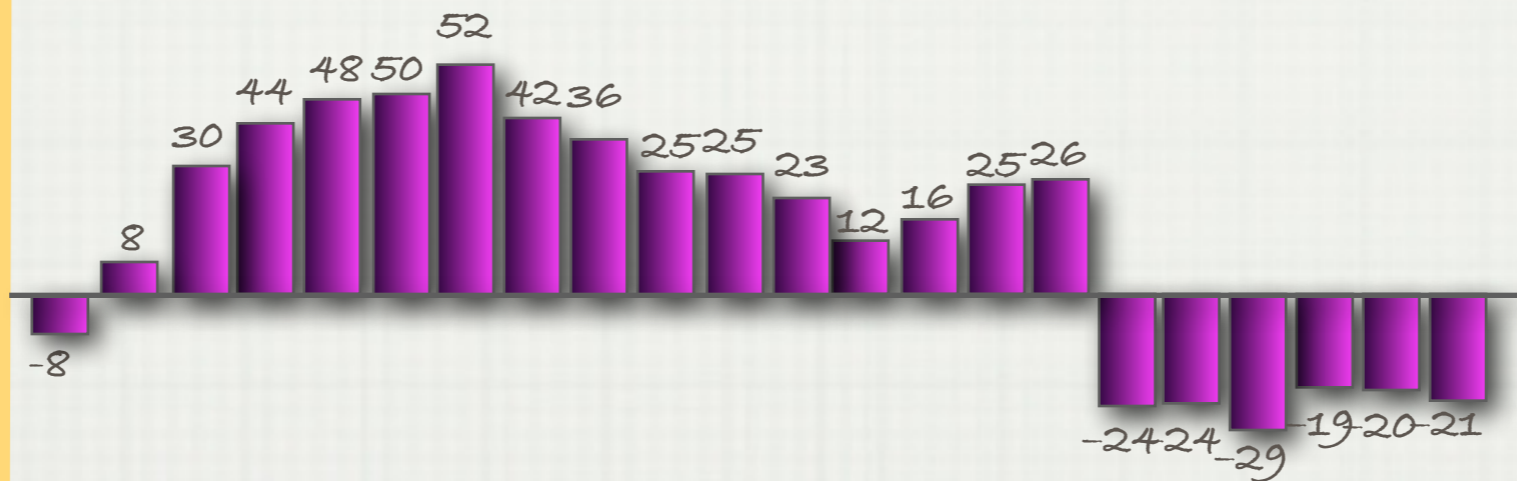
AALLOKEMUUNNOS



# KOHINANPOISTO

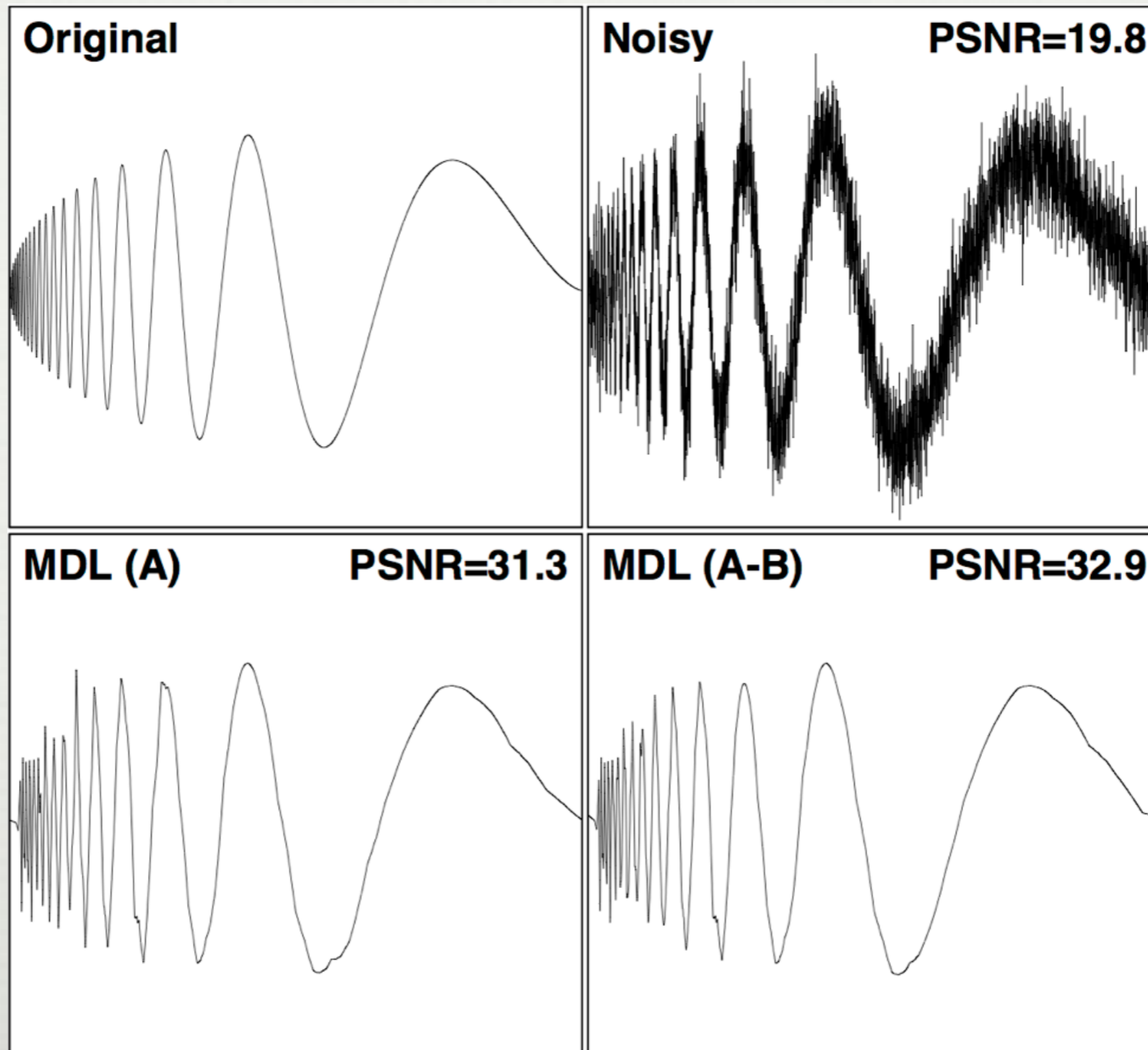
AALLOKEMUUNNOS

SIGNAALI "SILEÄMPI" KUIN ENNEN  
JA VÄHEMMÄN KOHINAA



"LUONNOLLINEN" SIGNAALI

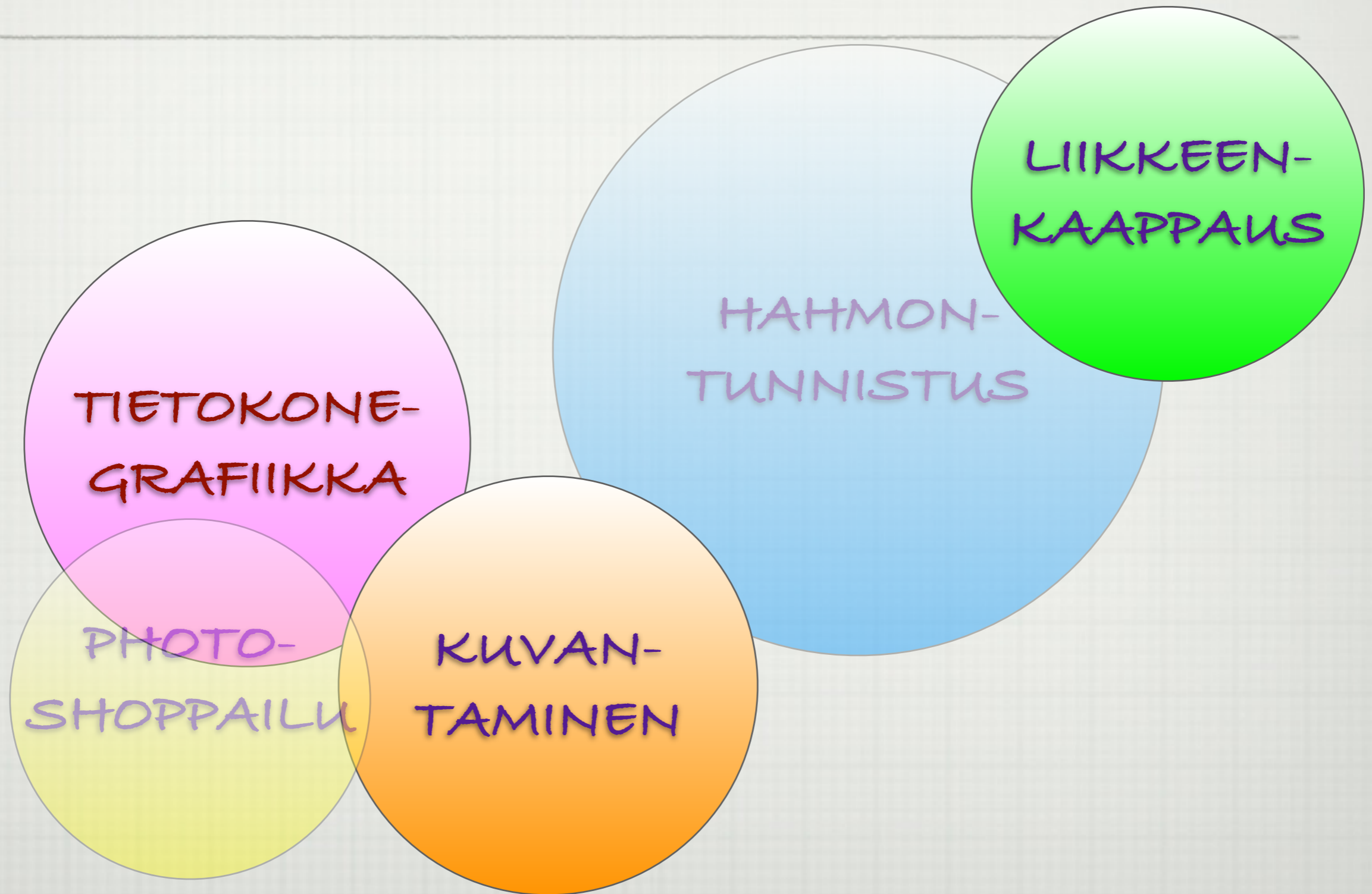
# KOHINANPOISTO





# SIGNAALINKÄSITTELY

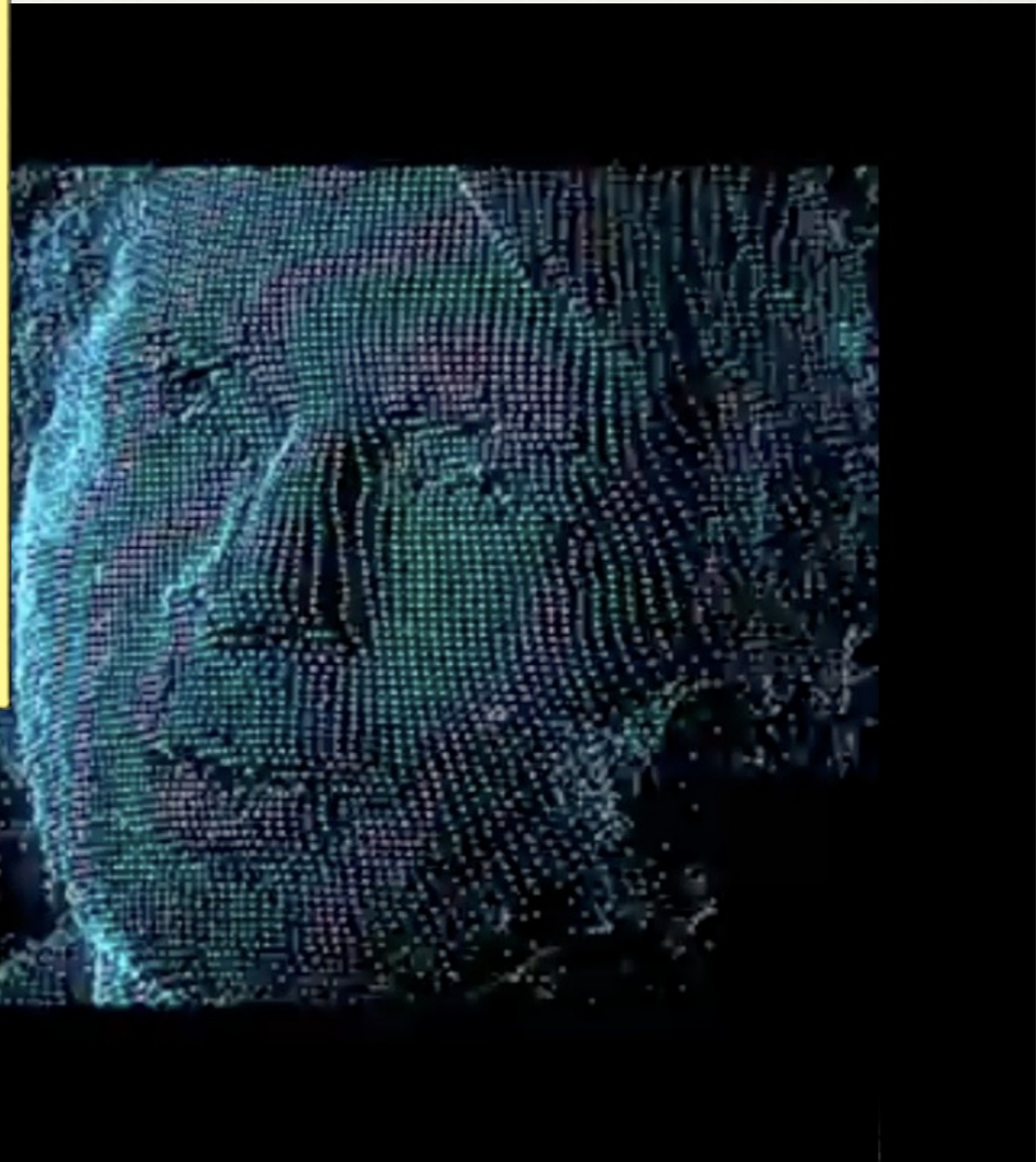
---



# RADIOHEAD

## LIDAR:

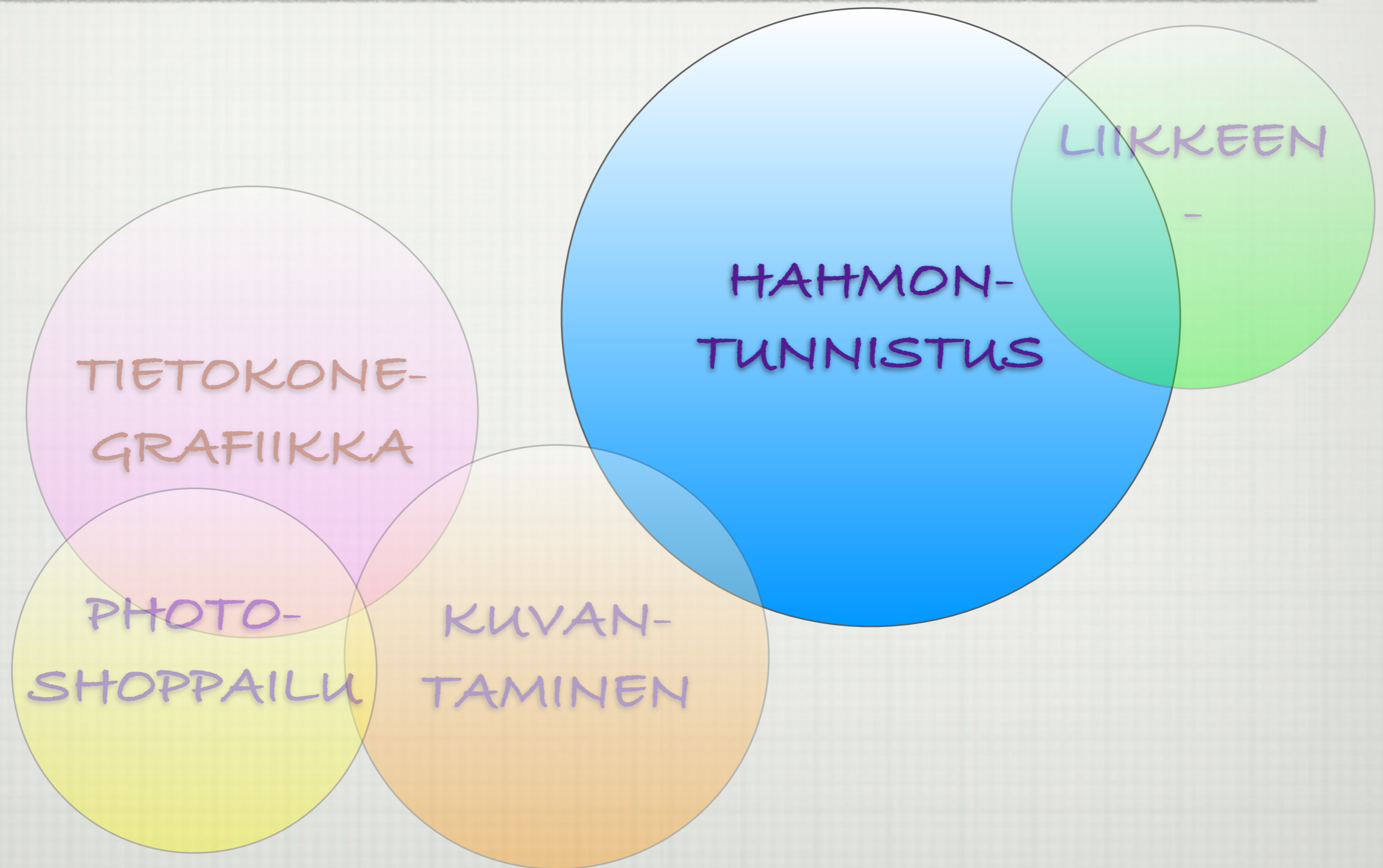
- MATALATEHOINEN LASERVALO HEIJASTUU KOHTEESTA.
- HEIJASTUMA KERTOO ETÄISYYDEN
- TULOKSENA SYVYYSKUVA
- SAMANTYYPPIINEN IDEA KÄYTÖSSÄ KINECTISSÄ!





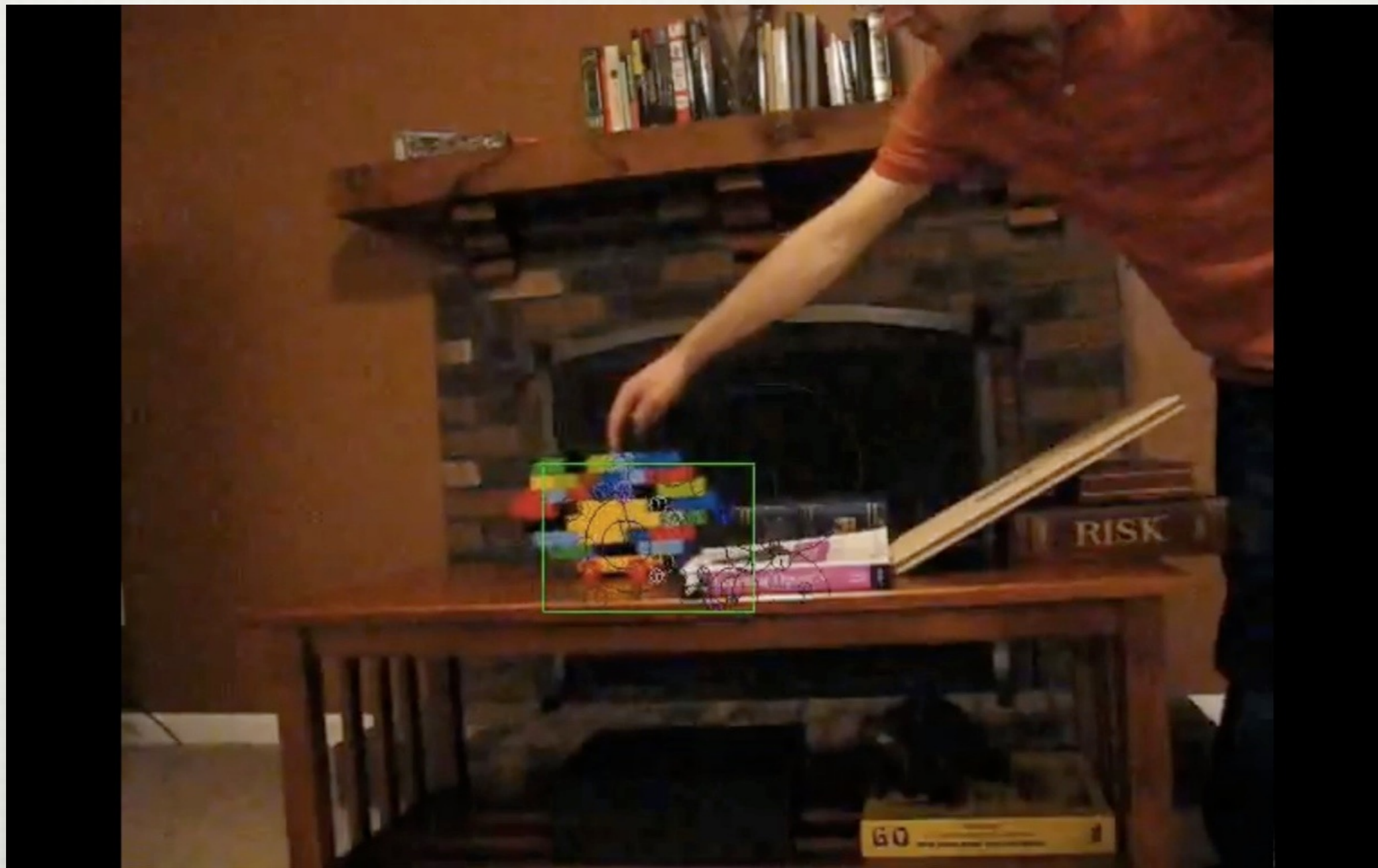
# SIGNAALINKÄSITTELY

---



# SEURANTA (TRACKING)

---





# HAHMONTUNNISTUS

---

- \* TAVOITE TUNNISTAMISESSA LÖYTÄÄ PIIRTEET, JOIDEN PERUSTEELLA VOIDAAN TUNNISTAA SAMA HAHMO ERI KUVISSA TAI ÄÄNINÄYTTEISSÄ
- \* KUVASSA TYYPILLISIÄ PIIRRETYYPPEJÄ:
  - REUNAT
  - KULMAT
- \* ÄÄNESSÄ:
  - TAAJUUS
  - TAAJUUDEN MUUTOKSET (YLÖS, ALAS, ...)
  - RYTMİ
  - ...

# SURF

---

- \* HAHMONTUNNISTUKSESSA SUOSIOSSA "INVARIANTIT PIIRTEET", KUTEN
  - \* **SIFT** (Scale Invariant Feature Transform)
  - \* **SURF** (Speeded Up Robust Features).
- \* H. Bay, T. Tuytelaars & L. van Gool. "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), vol. 110, No. 3, pp. 346-359, 2008



# SURF

---

- \* IDEANA LÖYTÄÄ KUVASTA JOUKKO PIIRTEITÄ (FEATURE), JOTKA SÄILYVÄT SAMANA
  - ERI KOOSSA
  - ERI KULMASSAJA JOTKA VOI LASKEA **INVARIANSSI**
  - NOPEASTI.
- \* TOISAALTA PIIRTEIDEN TULISI OLLA "YKSILÖLLISIÄ"
- \* TASAISEN PINNAN PISTE **EI OLE** YKSILÖLLINEN
- \* REUNA **EI OLE** YKSILÖLLINEN
- \* KULMA **ON** YKSILÖLLINEN

# SURF

---

## VAIHE 1: AVAINPISTEIDEN VALINTA:

1. ESITETÄ KUVA "INTEGRAALIMUODOSSA" (NOPEUTUS)

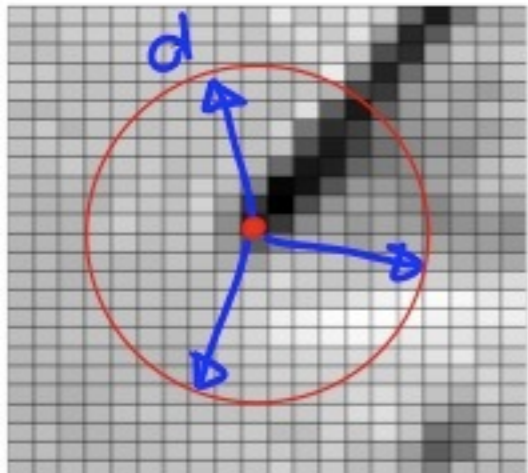
$$I(x, y) = \sum_{i=1}^x \sum_{j=1}^y F(i, j)$$

2. LASKE "HESSEN MATRIISIN (HESSIAN) DETERMINANTTI":  
INTENSITEETIN 2. DERIVAATAT ERI SUUNTIIN



# SURF

The Hessian matrix determines how  $S(x,y)$  will change from a unit-length displacement  $d$  in a given direction



$$S(x,y) \approx$$

$$S(0,0) + \text{Hessian matrix } H$$

$$\frac{1}{2} \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} \frac{\partial^2 S}{\partial x^2} & \frac{\partial^2 S}{\partial x \partial y} \\ \frac{\partial^2 S}{\partial x \partial y} & \frac{\partial^2 S}{\partial y^2} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

''':

# SURF

---

## VAIHE 1: AVAINPISTEIDEN VALINTA:

1. ESITETÄ KUVA "INTEGRAALIMUODOSSA" (NOPEUTUS)

$$I(x, y) = \sum_{i=1}^x \sum_{j=1}^y F(i, j)$$

2. LASKE "HESSEN MATRIISIN (HESSIAN) DETERMINANTTI":  
INTENSITEETIN 2. DERIVAATAT ERI SUUNTIIN
3. TOISTA ERI SKAALAUKSILLA (SKAALAINVARIANTTI)
4. VALITSE  $\det(H)$ :N PAIKALLISET MAKSIMIT AVAINPISTEIKSI



VAIHE 1: A...

1. ESITETÄ K...

$$I(x,y) =$$

2. LASKE "HE...  
INTENSITE...

3. TOISTA ER...

4. VALITSE d...



NOPEUTUS)

TERMINANTTI":  
TIIN

VARIANTTI)

AVAINPISTEIKSI

# SURF

---

## VAIHE II: PIIRTEIDEN KUVAUS:

1. TARKASTELE JOKAISEN AVAINPISTEEN YMPÄRISTÖÄ
2. LASKE ORIENTAATIO (INTENSITEETIN PERUSTEELLA)
3. KONSTRUOI INTENSITEETIN VAIHTELUN PERUSTEELLA KUVAAJAVEKTORI (SURFISSA 64-DIMENSIOINEN)



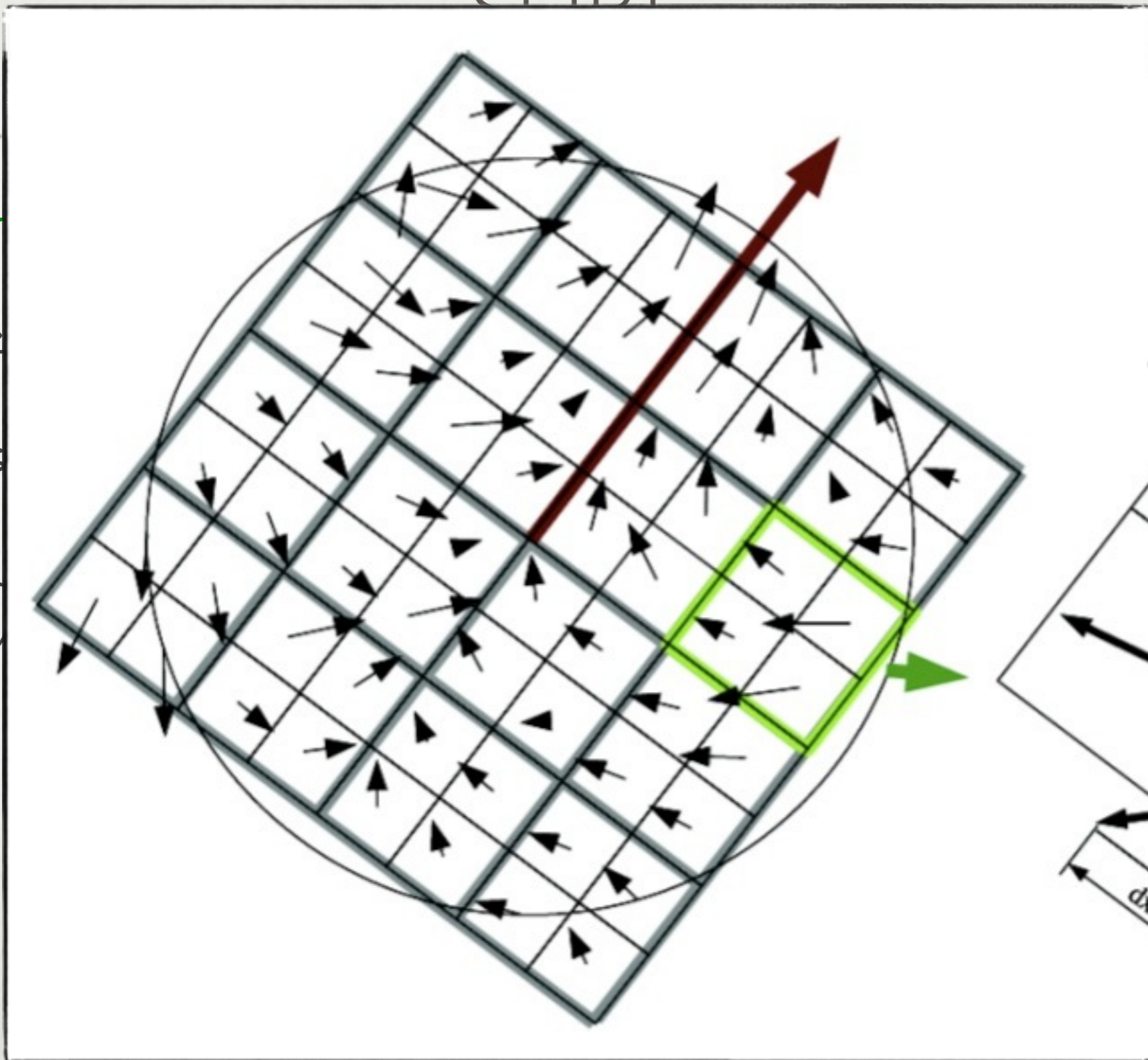
# CLIFF

VAIH

1. TAR

2. LAS

3. KON  
KUV



ÖÄ

LA)

ELLA

# SURF

---

## VAIHE II: PIIRTEIDEN KUVAUS:

1. TARKASTELE JOKAISEN AVAINPISTEEN YMPÄRISTÖÄ
2. LASKE ORIENTAATIO (INTENSITEETIN PERUSTEELLA)
3. KONSTRUOI INTENSITEETIN VAIHTELUN PERUSTEELLA KUVAAJAVEKTORI (SURFISSA 64-DIMENSIOINEN)

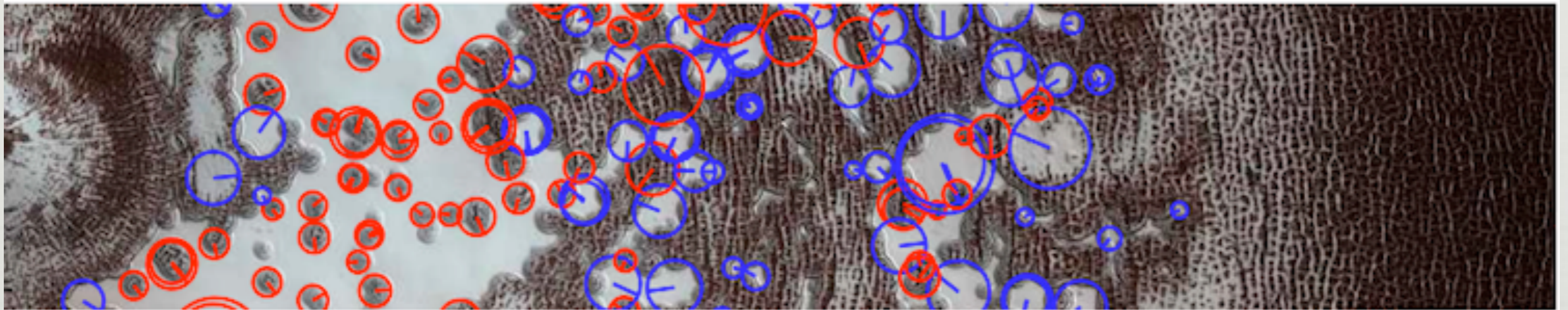
TULOKSENA PIIRREVEKTORI:

(X, Y, SKAALA, ORIENTAATIO, KUVAAJAVEKTORI)



# SURF

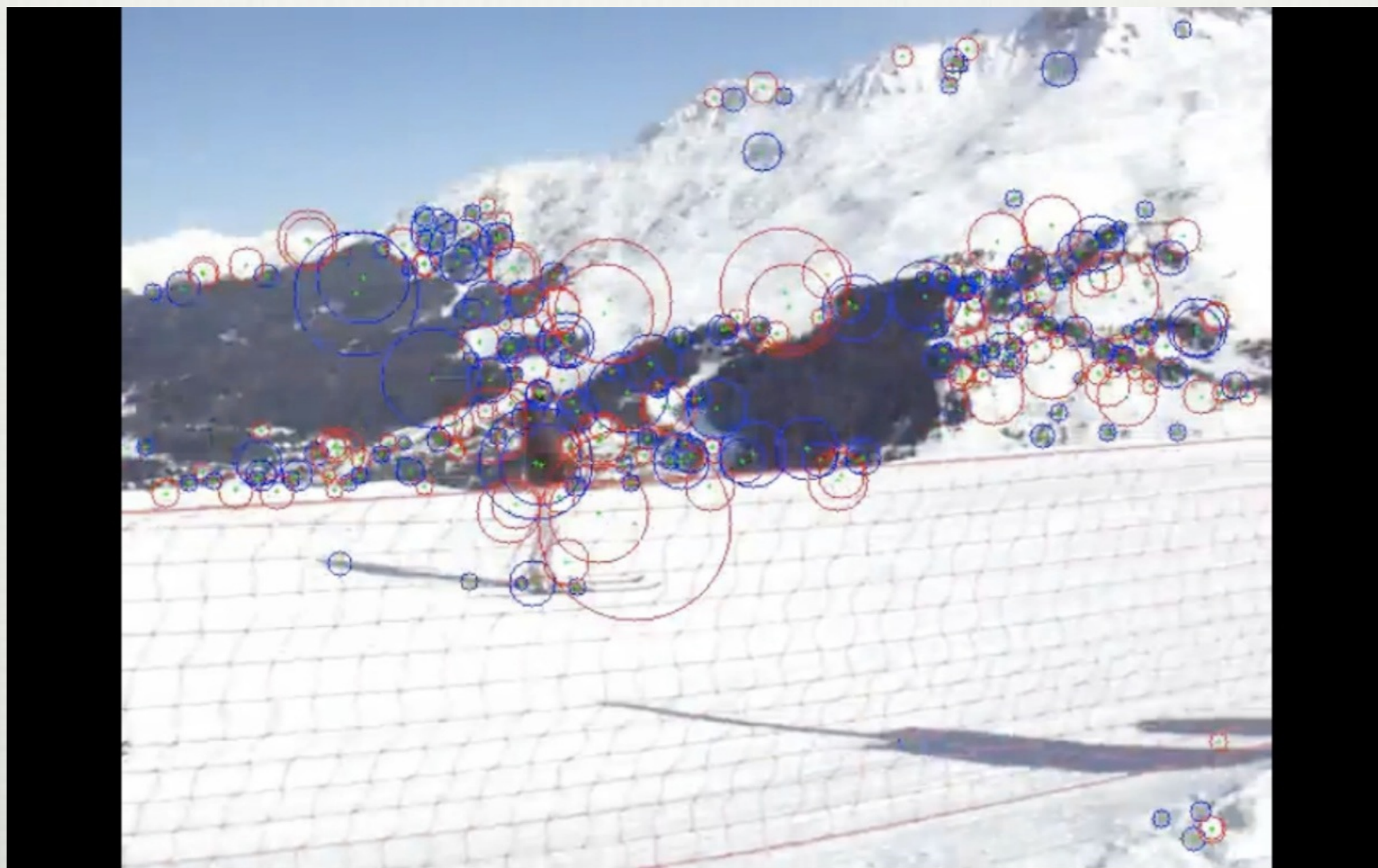
---





# SURF-ESIMERKKI

---





# SURF

---

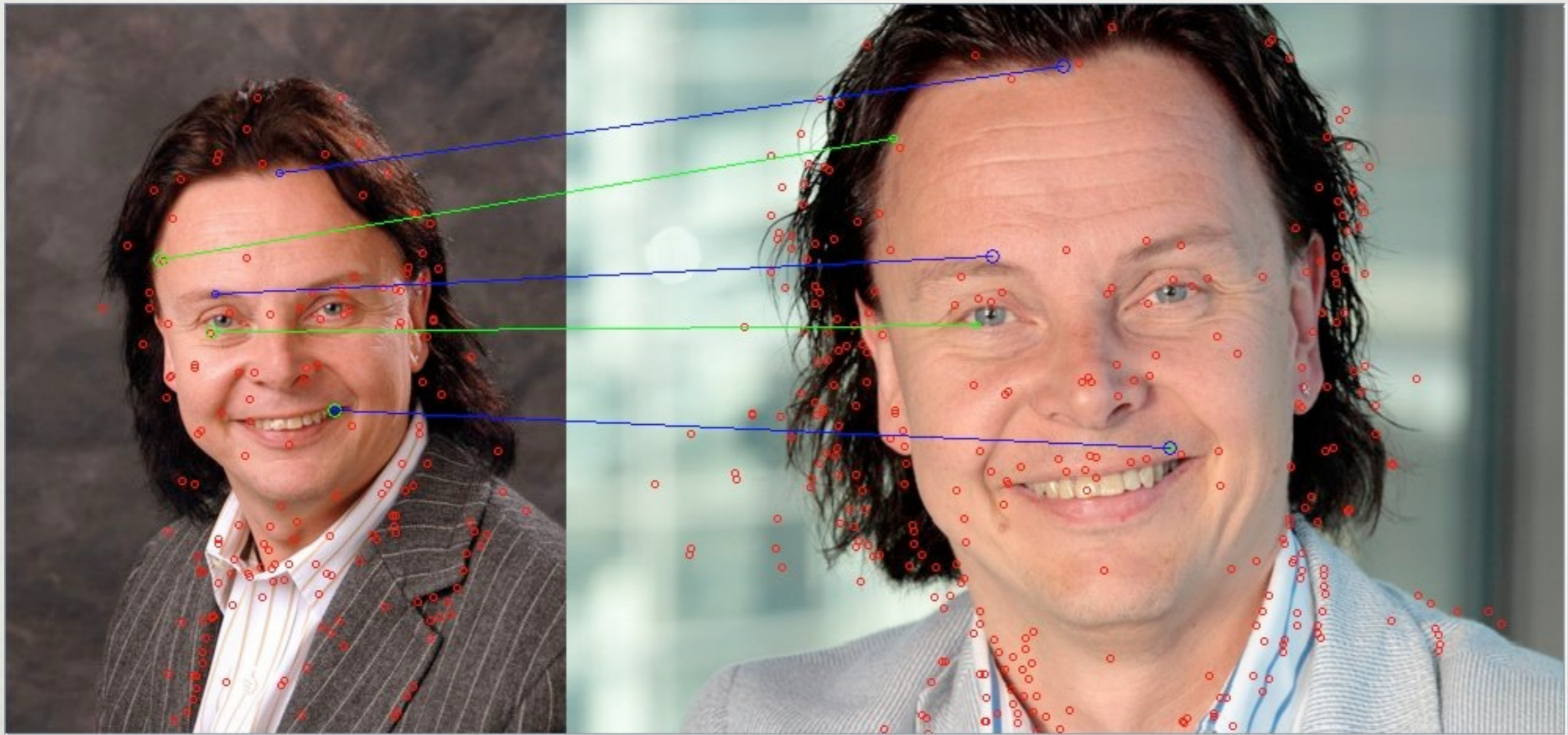
## VAIHE III: HAHMONTUNNISTUS

1. ETSI PIIRTEET YHDESTÄ KUVASTA
2. ETSI PIIRTEET TOISESTA KUVASTA
3. ETSI ERI KUVISSA ESIINTYVIÄ PIIRREPAREJA, JOTKA OVAT RIITTÄVÄN LÄHELLÄ TOISIAAN (EUKLIDINEN ETÄISYYS)
4. VOI PARANTAA GEOMETRISILLA RAJOITTEILLA (KORVAT ERI PUOLELLA PÄÄTÄ, SILMÄT SIINÄ VÄLISSÄ, JNE.)



# SURF

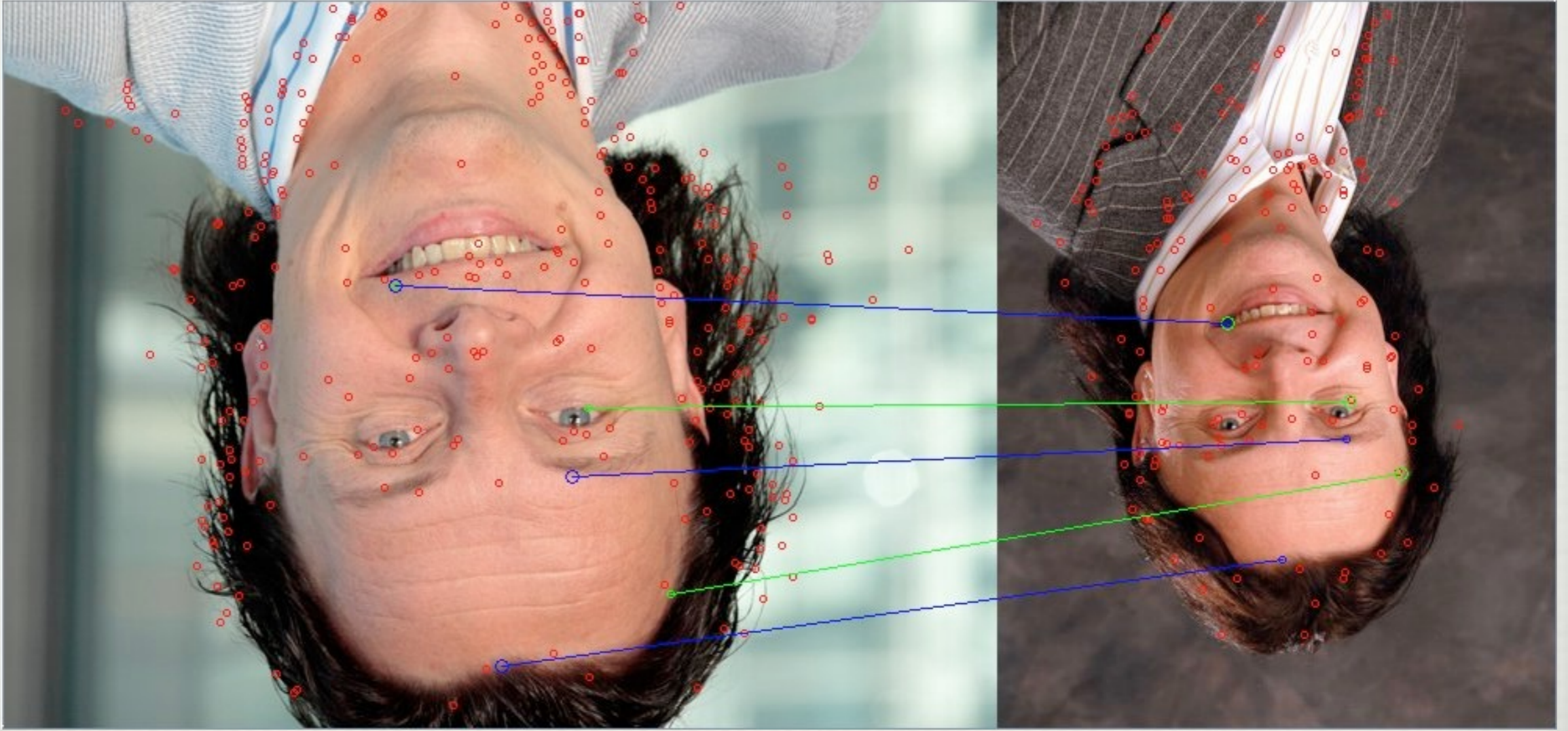
---





# SURF

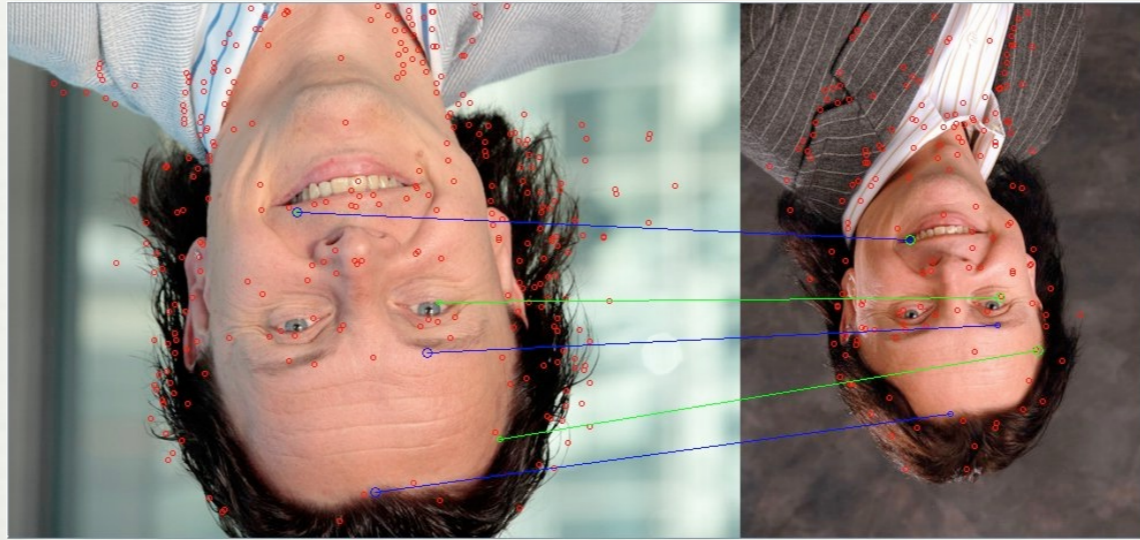
---





# SURF

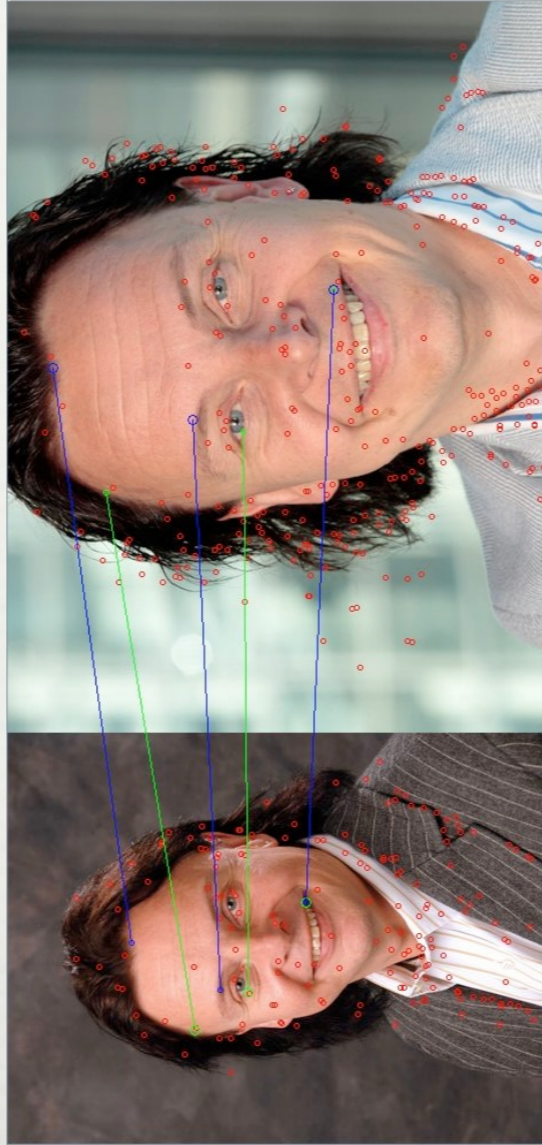
---





# SURF

---



# SURF

---





