

Detekcija objekata: SLAMovozeći automobili

Kosta Grujčić

Matematička gimnazija

NEDELJA^{v5.0}
INFORMATIKE

18. decembar 2018.

Podsećanje



Podsećanje



► Integral

Podsećanje



- ▶ Integral
- ▶ Verovatnoća
 - ▶ slučajne promenljive
 - ▶ uslovna verovatnoća
 - ▶ raspodela i zajednička raspodela

Motivacija



- ▶ Želimo da se automobil *sam* kreće kroz prostor
- ▶ Želimo da automobil *zna* šta je oko njega

Motivacija



- ▶ Želimo da se automobil *sam* kreće kroz prostor
- ▶ Želimo da automobil *zna* šta je oko njega
- ▶ Radi jednostavnosti, automobil smatramo *agentom*

Prostor i kretanje



- ▶ Kompozicija translacija i rotacija
 - ▶ afine transformacije

Prostor i kretanje



- ▶ Kompozicija translacija i rotacija
 - ▶ afine transformacije
- ▶ Euklidski prostor *nije* dobar model prostora
 - ▶ Projektivna geometrija

Šta je SLAM?



- ▶ Određivanje okruženja – *mapiranje*
- ▶ Određivanje pozicije agenta – *lokalizacija*

Šta je SLAM?



- ▶ Određivanje okruženja – *mapiranje*
- ▶ Određivanje pozicije agenta – *lokalizacija*
- ▶ **Istovremeno!**

Formalna postavka



- ▶ \mathbf{o}_t – opservacije u trenutku t
- ▶ \mathbf{c}_t – radnja koju agent preduzima u trenutku t
- ▶ \mathbf{x}_t – pozicija agenta u trenutku t
- ▶ \mathbf{m} – mapa okruženja u trenutku

- ▶ Treba izračunati $P(\mathbf{x}_t, \mathbf{m} | \mathbf{o}_{1:t}, \mathbf{c}_{1:t})$
 - ▶ *online* SLAM

Zašto je teško?



- ▶ Pozicija robota nam treba za mapu okruženja
- ▶ Mapa okruženja nam treba za poziciju robota
- ▶ Neizbežan šum prilikom opažanja
- ▶ Opservacije su potpuno neprecizne na duže staze

Kalmanov filter



Kalmanov filter



- ▶ Pretpostavlja da je model linearan

Kalmanov filter



- ▶ Pretpostavlja da je model linearan
- ▶ Pretpostavlja da je šum normalne raspodele

Kalmanov filter



- ▶ Pretpostavlja da je model linearan
- ▶ Pretpostavlja da je šum normalne raspodele
- ▶ Rekurzivan postupak
 - ▶ *estimation step*
 - ▶ *update step*
 - ▶ za svaki korak je neophodan samo prethodni

Linearnost modela



- ▶ $\mathbf{x}_t = A_t \mathbf{x}_{t-1} + B_t \mathbf{c}_t + \epsilon_t$
- ▶ $\mathbf{o}_t = C_t \mathbf{x}_t + \delta_t$

- ▶ A_t – matrica **ekskluzivne** veze trenutne i prethodne pozicije
- ▶ B_t – matrica **ekskluzivne** veze trenutne pozicije i trenutne radnje
- ▶ C_t – matrica veze trenutne pozicije i trenutne opservacije
- ▶ ϵ_t, δ_t – trenutni šum normalne raspodele kretanja i opservacija redom
 - ▶ kovarijacije su redom E_t i D_t

Algoritam



KALMANFILTER($\boldsymbol{\mu}_{t-1}$, Σ_{t-1} , \mathbf{c}_t , \mathbf{o}_t)

1. $\hat{\boldsymbol{\mu}}_t = A_t \boldsymbol{\mu}_{t-1} + B_t \mathbf{c}_t$
2. $\hat{\Sigma} = A_t \Sigma_{t-1} A_t^T + E_t$
3. $K_t = \hat{\Sigma}_t C_t^T (C_t \hat{\Sigma}_t C_t^T + D_t)^{-1}$
4. $\boldsymbol{\mu}_t = \hat{\boldsymbol{\mu}}_t + K_t (\mathbf{o}_t - C_t \hat{\boldsymbol{\mu}}_t)$
5. $\Sigma_t = (I - K_t C_t) \hat{\Sigma}_t$
6. *return* $\boldsymbol{\mu}_t$, Σ_t

Nelinearan model



- ▶ Matrice A_t , B_t i C_t se menjaju odgovarajućim vektorskim funkcijama

Nelinearan model



- ▶ Matrice A_t , B_t i C_t se menjaju odgovarajućim vektorskim funkcijama
- ▶ Ivršićemo lokalnu linearizaciju
 - ▶ Tejlorov razvoj do prvog člana
 - ▶ Lokalno aproksimiramo tangentni prostor

Nelinearan model



- ▶ Matrice A_t , B_t i C_t se menjaju odgovarajućim vektorskim funkcijama
- ▶ Ivršićemo lokalnu linearizaciju
 - ▶ Tejlorov razvoj do prvog člana
 - ▶ Lokalno aproksimiramo tangentni prostor
- ▶ Zahteva više izračunavanja

EKF SLAM



$$\mu = \begin{bmatrix} \mathbf{x} \\ \mathbf{m} \end{bmatrix} \quad \Sigma = \begin{bmatrix} \Sigma_{xx} & \Sigma_{xm} \\ \Sigma_{mx} & \Sigma_{mm} \end{bmatrix}$$

- ▶ $\mathbf{x} = (x, y, \theta_x, \theta_y, \theta_z)^T$
- ▶ $\mathbf{m} = (m_{1,x}, m_{1,y}, \dots, m_{n,x}, m_{n,y})^T$
- ▶ $(\Sigma_{xx})_{i,j} = \text{COV}[\mathbf{x}_i, \mathbf{x}_j]$
- ▶ $(\Sigma_{xm})_{i,j} = (\Sigma_{mx})_{j,i} = \text{COV}[\mathbf{x}_i, \mathbf{m}_j]$
- ▶ $(\Sigma_{mm})_{i,j} = \text{COV}[\mathbf{m}_i, \mathbf{m}_j]$

Plan rada



1. Procena stanja
2. Procena opažanja
3. Opažanje
4. Asocijacija podataka
5. Ažuriranje

Šta vidi SLAM?



- ▶ SLAM ne razlikuje psa od kamiona
- ▶ Potrebna nam je *inteligencija*
- ▶ **Mora** da radi u realnom vremenu
- ▶ Hardverska zahtevnost **mora** biti što je moguće manja

Podsećanje



Podsećanje



- ▶ Neuronske mreže
 - ▶ funkcije aktivacije
 - ▶ propagacija unazad
 - ▶ konvolucija

YOLO algoritam



YOLO algoritam



- ▶ Vršiti detekciju objekata

YOLO algoritam



- ▶ Vršiti detekciju objekata
- ▶ Određuje okvir svakog objekta

YOLO algoritam

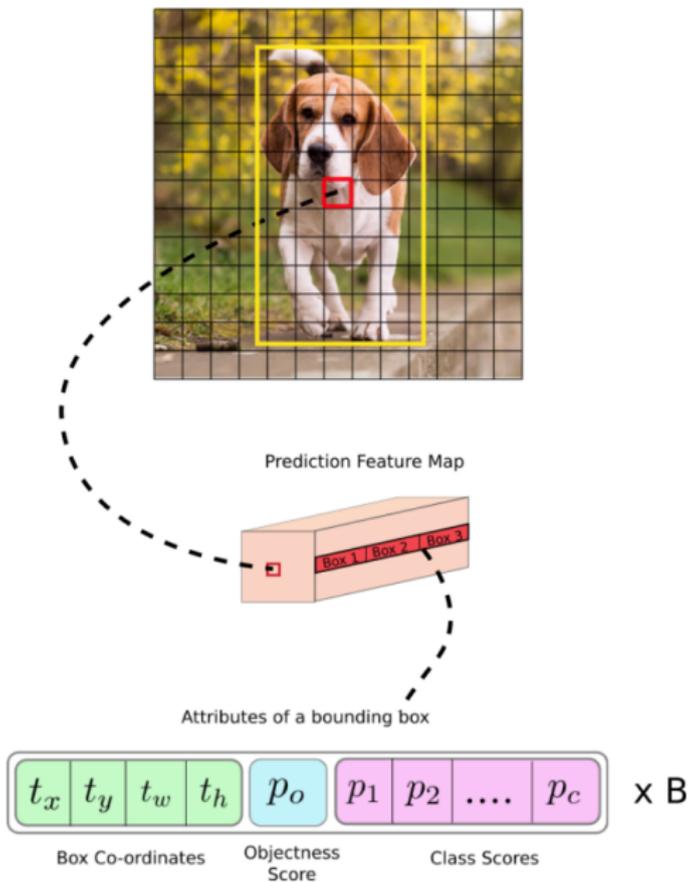


- ▶ Vršiti detekciju objekata
- ▶ Određuje okvir svakog objekta
- ▶ Veoma efikasan – koristi samo jedan *CNN*

Kako YOLO radi?



- ▶ Slika se deli na *manje* kvadrate – ćelije
- ▶ Ćelija koja je u centru objekta ga određuje
- ▶ Svaka ćelija ima svoj okvir – *bounding box*
- ▶ Za svaku ćeliju se određuju verovatnoće za svaku klasu



Okviri

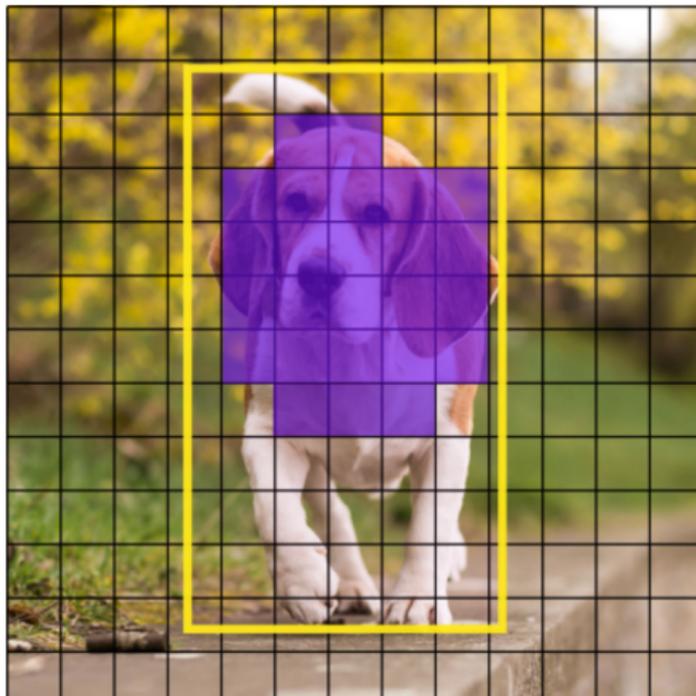


- ▶ Za svaku ćeliju po B okvira
- ▶ Određen centrom, dužinom i širinom
- ▶ Relativna reprezentacija

Duplikati

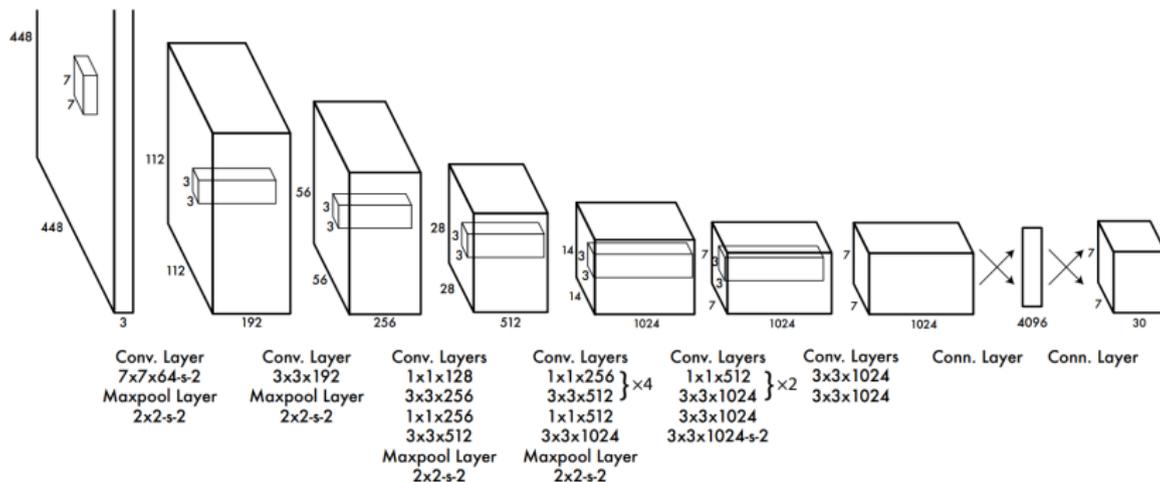


- ▶ Svaki okvir je dovoljno *dobar* – trivijalno
- ▶ Susedne ćelije mogu detektovati isti objekat – problem
 - ▶ Rešenje: *Non-maximum suppression*

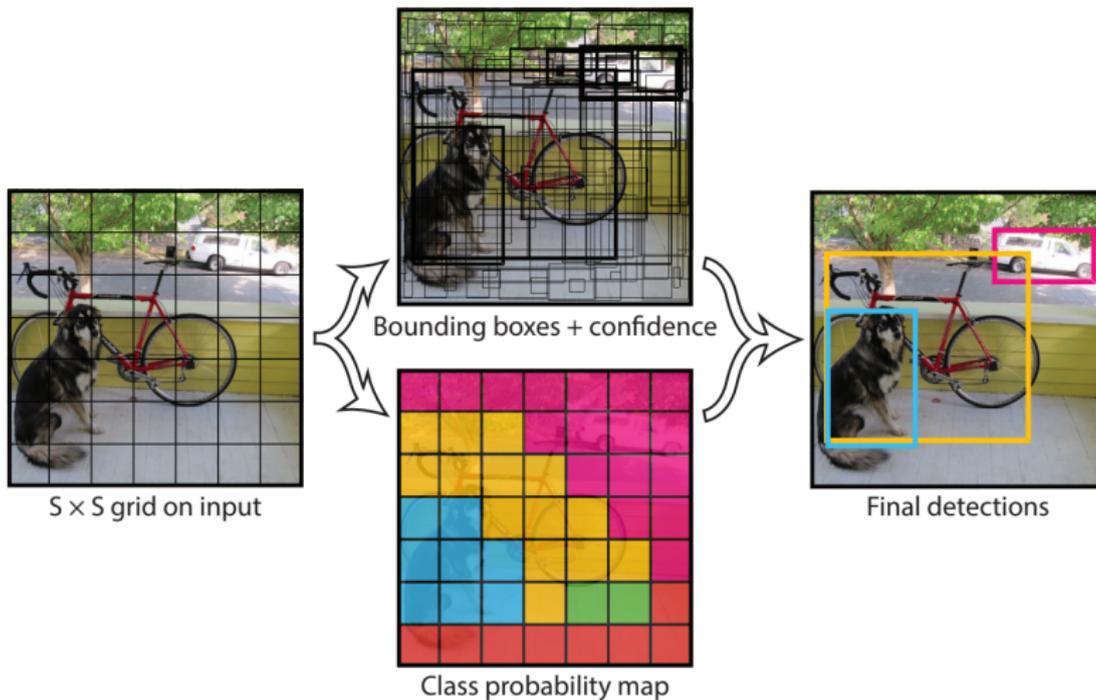


Slika: Žuti okvir – *ground-truth*

Arhitektura mreže



Pipeline



Rekapitulacija



- ▶ Agent *zna* gde je u prostoru
- ▶ Agent je *svestan* prostora oko sebe
- ▶ Agent *zna* šta se oko njega nalazi