



# Handling Occlusions in Augmented Reality

*Projektevaluering den 17/6/2002*

10. Semester, Gruppe 1025

02gr1025@miba.auc.dk



# Program

1. Introduktion
2. Background subtraction
3. Visual hull
4. Post processering
5. Visualisering
6. Forbedringer



# Introduktion



# Motivation for projektet

Rekonstruktionssystem fra VISICS, K. U. Leuven

- Input:
  - Ukalibreret billedesekvens af statisk scene
- Output:
  - 3D model med tekstur
  - Kameraparametre ( $f_x, f_y, O_x, O_y, k, R, t$ )

Kan ikke direkte bruges til at håndtere okklusioner i augmented reality





# Testsekvens

Optaget i den botaniske have i Leuven, Belgien

- Optaget med håndholdt DV videokamera
- 37 billeder
- Kompleks forgrundsgeometri
- Camouflageeffekter

# Testsekvens og rekonstruktion



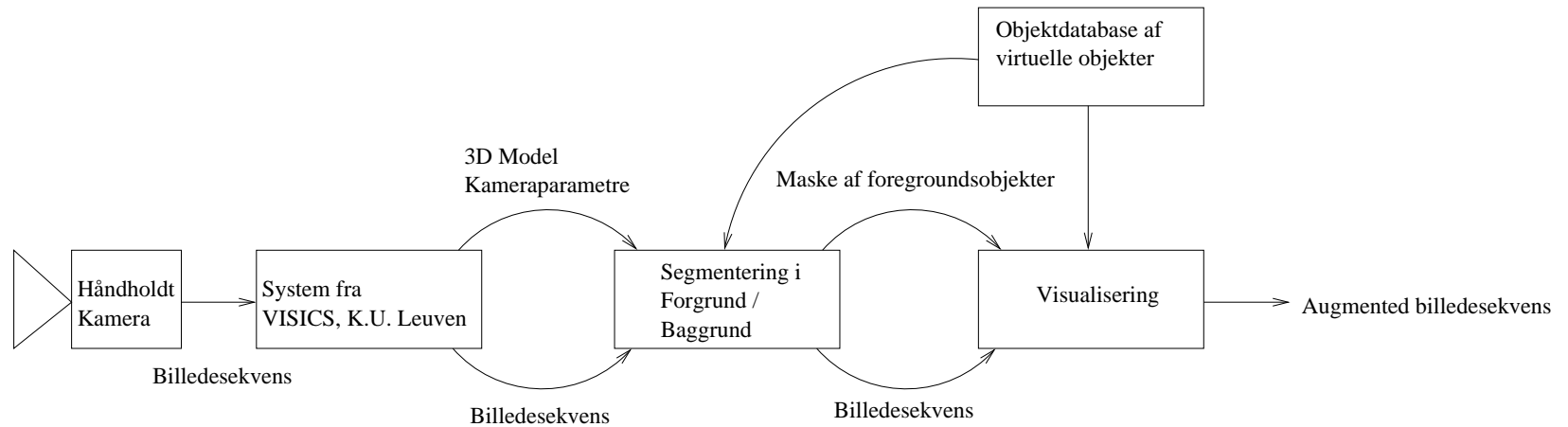
## Videsekvens

- Inputsekvens
- Rekonstruktion via depth map fra billede 10

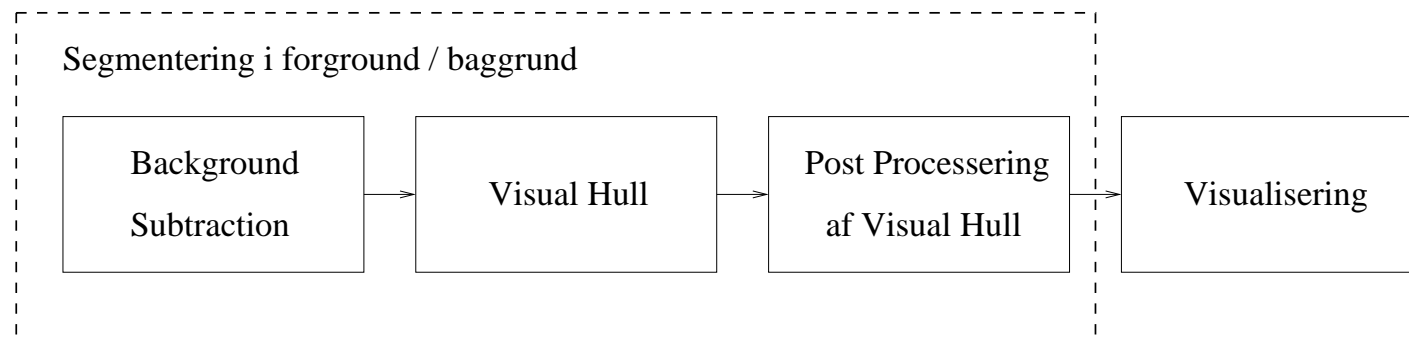


# Systembeskrivelse

## En udvidelse af rekonstruktionssystemet



## Består af 4 dele





# Background subtraction



# Formål

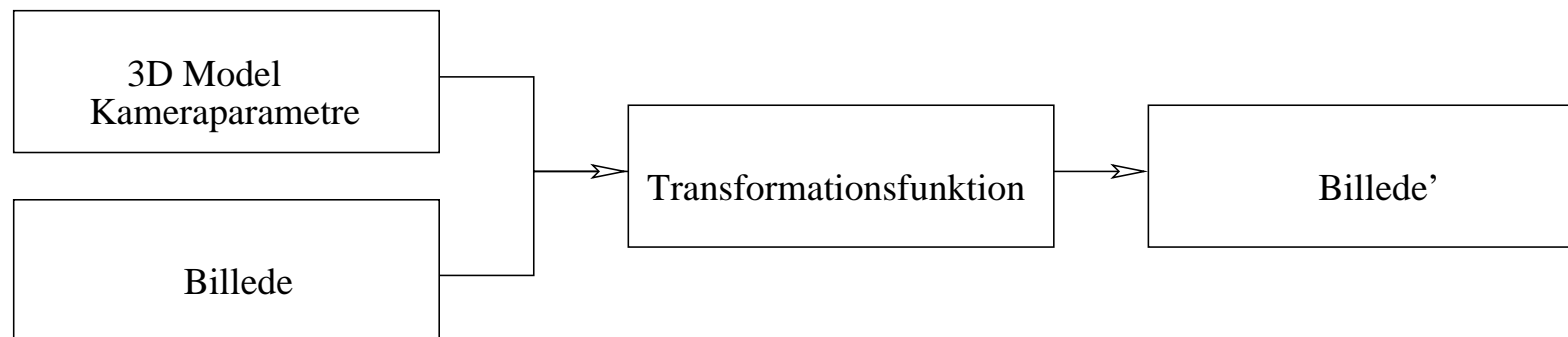
Beregne foreløbig okklusionsmaske ud fra farveforskel mellem forgrund og baggrund

- Til rådighed
  - 3D model
  - Kameraparametre for hvert view
  - Billede for hvert view
- Ide
  - Virtuel bluescreen
  - Model af baggrund beregnes ud fra billedesekvens
  - Farveforskel mellem forgrund og baggrund segmenteres

# Image warping



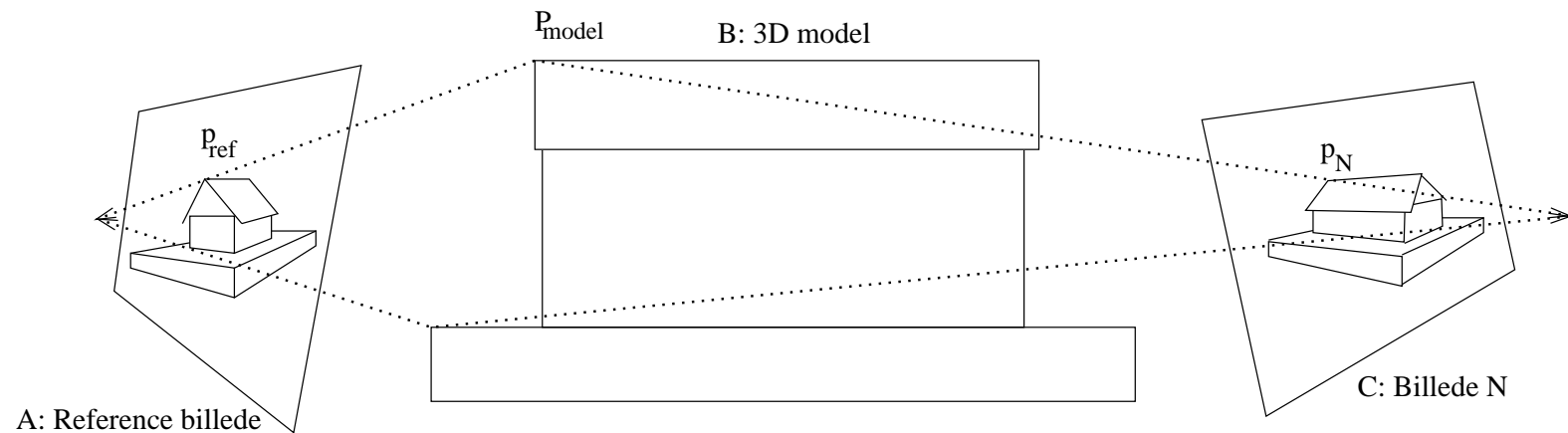
Background subtraction benyttes normalt i situationer med statisk kamera



Transformation af billeder til referencesynspunkt via 3D model og kameraparametre.



# Transformationsfunktion



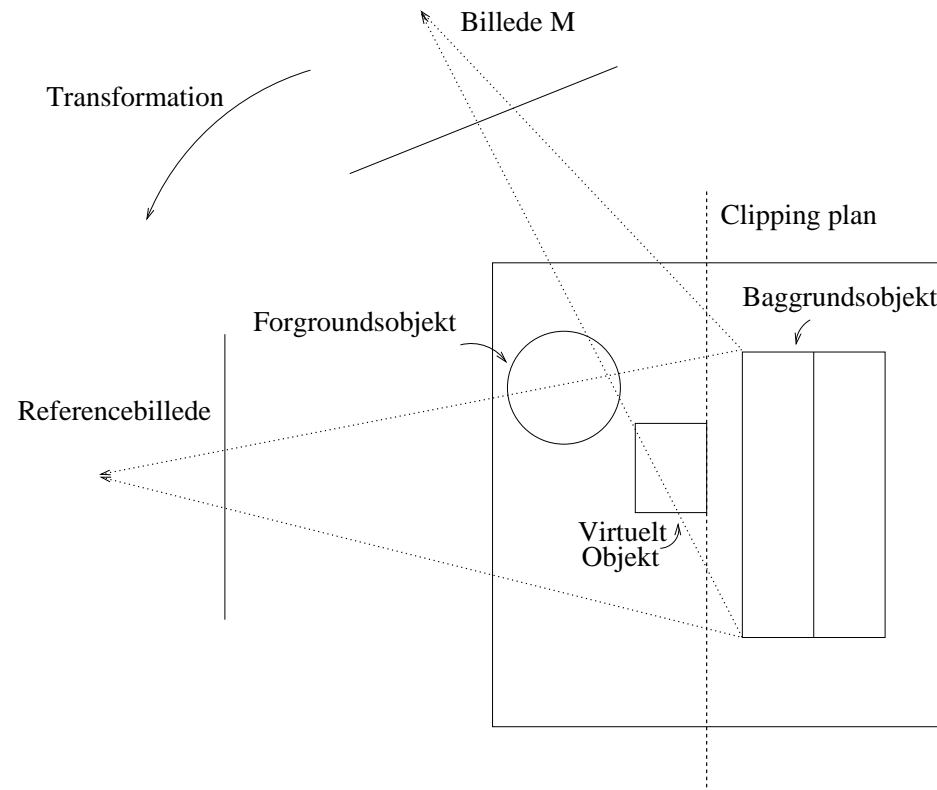
## Beregnes baglæns

- $p_{ref}$  backprojectes til 3D model
- $P_{model}$  projiceres til andet view
- $p_N$  svarer til  $p_{ref}$



# Clipping plan

Kun model af baggrunden



Okklusionsskygger er et problem



# Image warping



## Videsekvens

- Transformerer til billede 10
- Okklusionsskygger
- Manglende 3D rekonstruktion af forgrund



# Baggrundsmodel

Beskrivelse af distribution af baggrundsfarve for en given pixel

- Fejl i image warping
- Ændring i lysforhold

Modeller

- Middel
- Median

# Baggrundsmodel



## Videosekvens

- Median model
- Dækker kun en mindre del af baggrunden

# Segmentering



Segmentere i forgrund og baggrund

- Lav threshold klassificerer baggrund som forgrund
- Høj threshold forårsager huller på grund af camouflage

Metode

- Global threshold på farveforskel i RGB

# Segmentering



## Videosekvens

- Okklusionsmaske via depth map 10
- Okklusionsmaske via depth map fra nuværende view
- Okklusionsmaske via background subtraction



# Visual hull

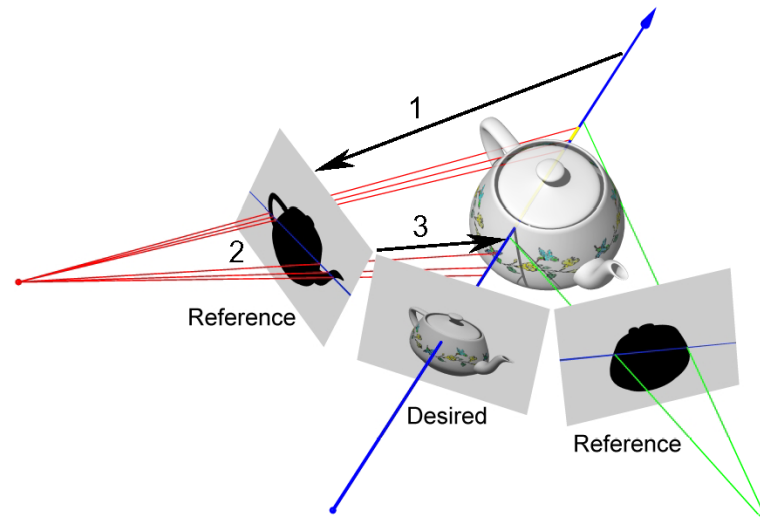


# Formål

Find manglende struktur i okklusionsmasken som er forårsaget af camouflageeffekter

- Til rådighed
  - 3D model
  - Kameraparametre for hvert view
  - Okklusionsmaske af forgrundsobjekter med camouflageeffekter
- Ide
  - Find 3D struktur som flere views er enige om baseret på okklusionsmaskerne for de enkelte views

# Princippet for visual hull



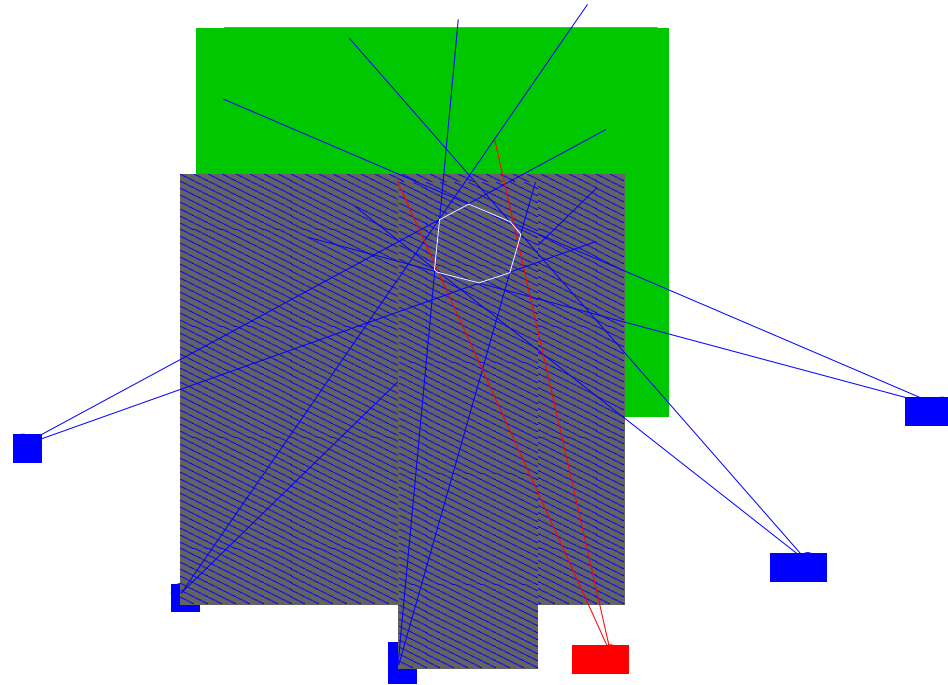
For alle pixels i det ønskede view

- 1) Projicer en 3D stråle ned på et reference view (epipolar linie)
- 2) Find liste af skæringsintervaller
- 3) Løft skæringsintervallerne tilbage på strålen





# Problemer med visual hull



- Bruger ikke alle views til at skære 3D strukturen med
- Lukker alle hullerne i okklusionsmasken
- Alle views skal være enig om 3D struktur



# Ny visual hull metode

Forskel i forhold til den generelle visual hull

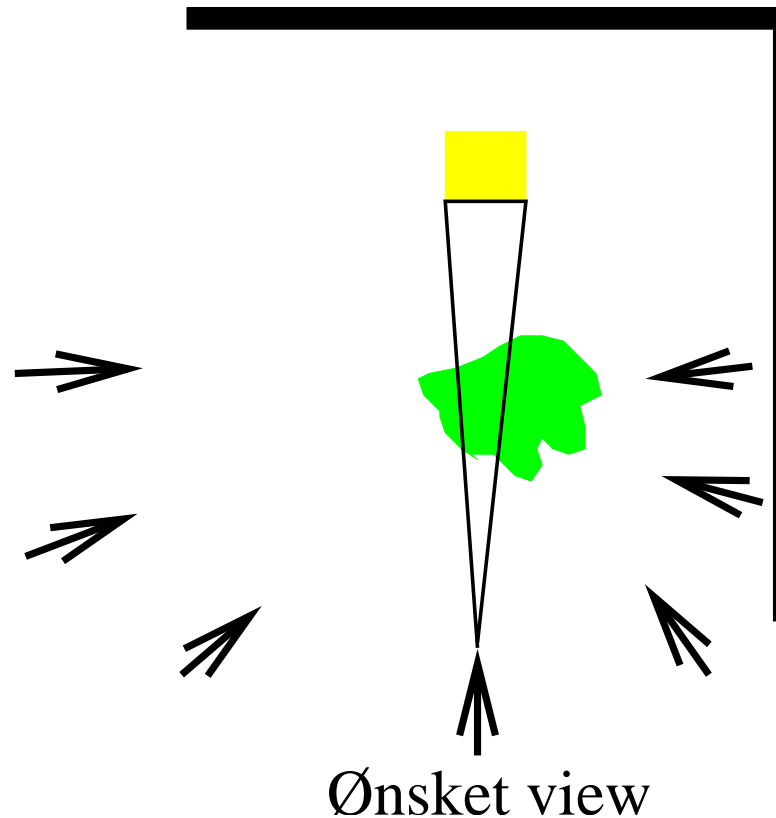
- Alle views mod alle andre views (fælles visual hull)
- Skyder kun stråler igennem okklusionsmasken
- Et antal views skal være enig om 3D struktur

Resultat af den nye metode

- Fælles visual hull for alle views
- Overflade visual hull

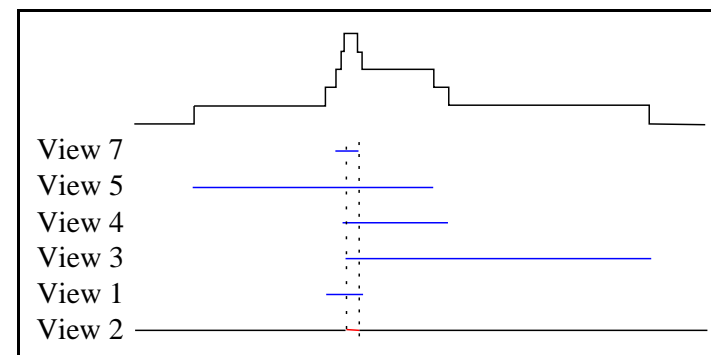
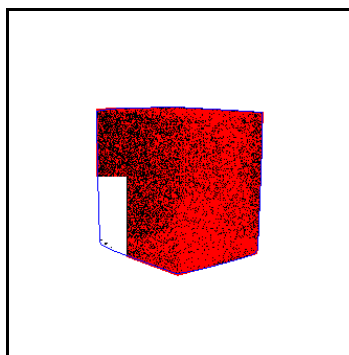
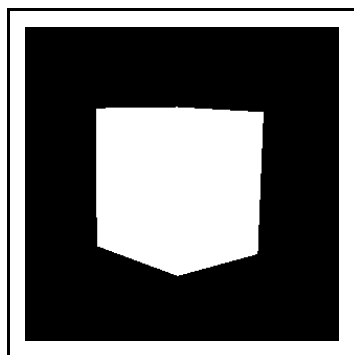
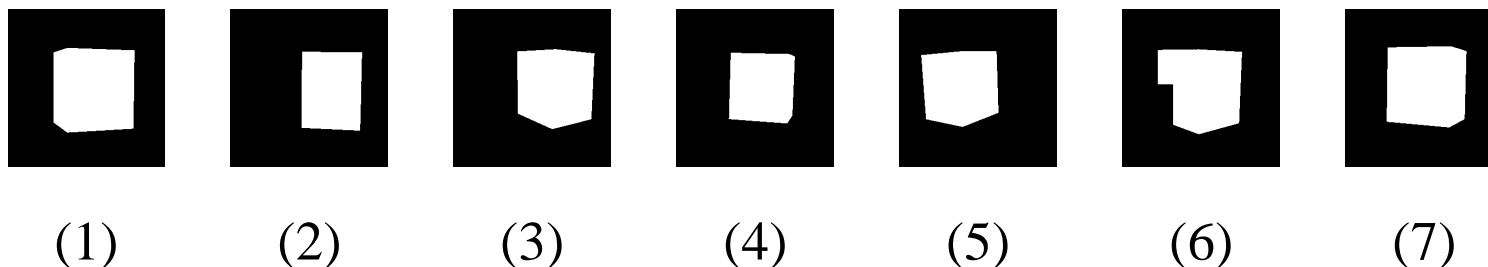


# Silhouetter af 3D strukturen



- Volumenet mellem det ønsket view og det virtuelle objekt
- Volumetric model af scenen med komplet 3D model

# Tilføj struktur



- Et minimum af views skal være enig
- Spidser i histogrammet betyder 3D struktur
- 3D strukturen tilføjes til okklusionsmasken ved projektion
- Overflade visual hull kombineret med convex hull lukker masken



# Post processing

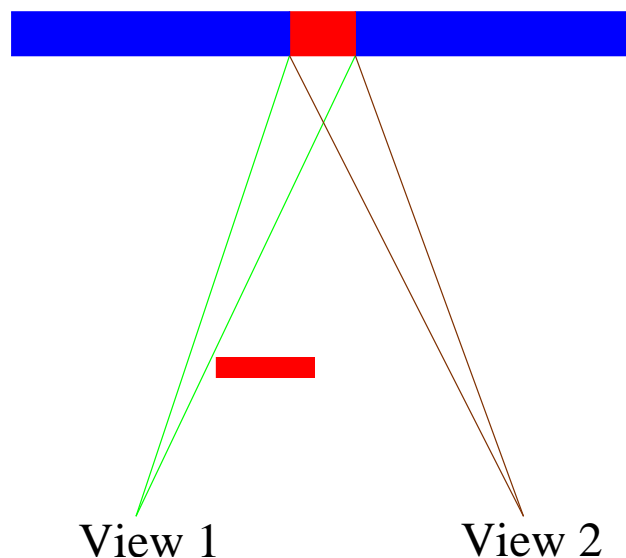


# Formål

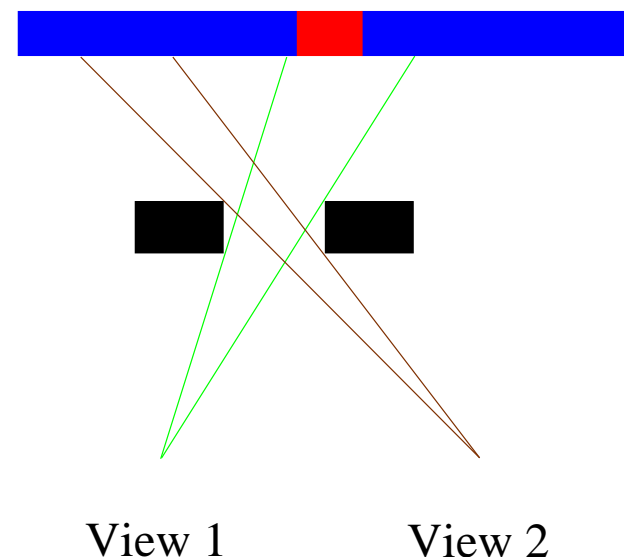
Luk camouflagede huller i okklusionsmasken

- Til rådighed
  - 3D model
  - Kameraparametre for hvert view
  - Forbedret okklusionsmaske af forgrundsobjekter
- Ide
  - Observeres et *rigtigt* hul fra forskellige vinkler vil mønsteret i hullet ændre sig i takt med at forskellige dele af baggrunden observeres gennem hullet

# Typer af huller



Camouflagehul



Rigtigt hul

**Camouflagehul:** Bevæger sig hen over silhouetten of objektet, samt der sker ingen ændring i det der observeres igennem hullet

**Rigtigt hul:** Forbliver på det samme sted i silhouette af objektet, samt der sker en ændring i det der observeres igennem hullet. Skæringen mellem viewene sker inde i objektet

# Forhåndsbestemmelse af huller



Hullerne i de enkelte silhouetter inddeles i fire grupper. Det gøres for at mindske kompleksiteten af den efterfølgende test

**Støj** Små huller som kan skyldes støj i okklusionsmasken lavet af background subtraction algorithmen. Fjernes fra okklusionsmasken

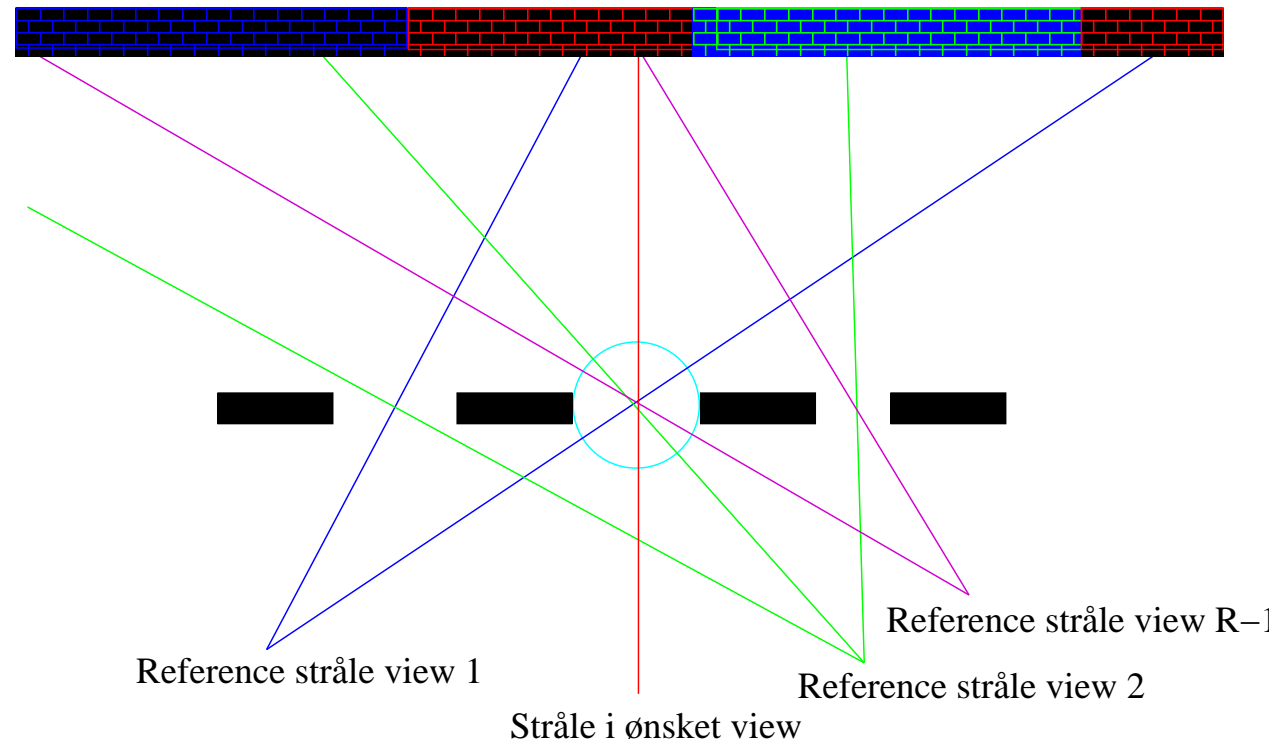
**Middel hul** Huller som er for store til at kaldes for støj og for små til at lave en statistik sammenligning. Bliver en blanding mellem forgrundsobjekt og virtuel objekt.

**Rigtigt hul** Huller hvor der findes en rekonstruktion af baggrunden. Forbliver i okklusionsmasken

**Potentiel hul** De resterende huller som testes for typen

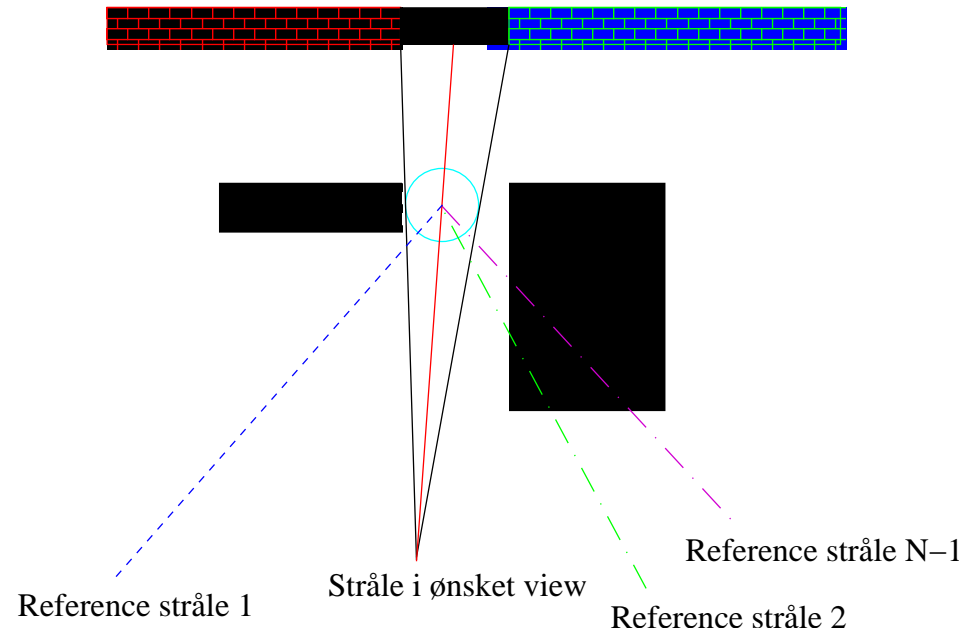


# Bestemmelse af typen



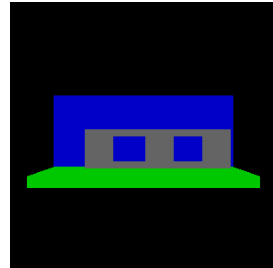
- Find skæringer inde i objektet
- Sammenlign det som observeres gennem hullet ved hjælp af farvefordeling

# Bestemmelse af typen

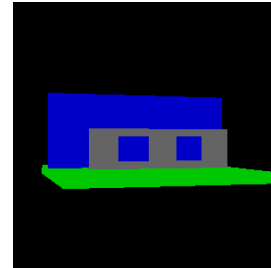


- Find et 3D punkt i midten af det potentionelle hul i forgrundsobjektet
- Find det sted i reference billedet som svarer til det potentionelle hul
- Sammenlign det som observeres gennem hullet ved hjælp af farvefordeling

# Eksempel på et camouflagehul



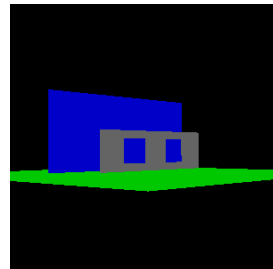
Ønsket view



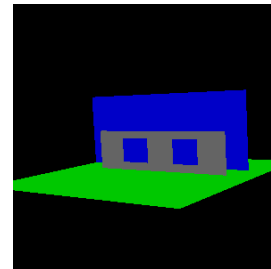
Reference view 1



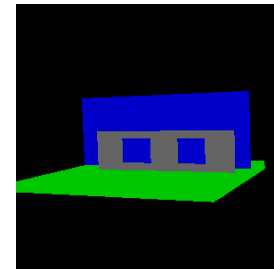
Reference view 2



Reference view 3



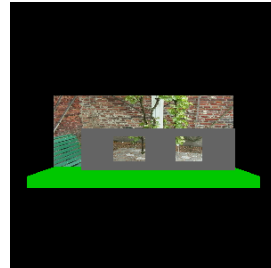
Reference view 4



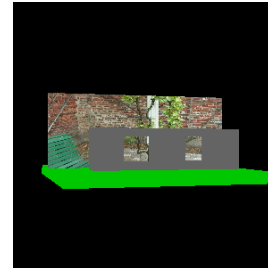
Reference view 5

1. De to potentionelle huller i forgrundsobjektet har samme farve som baggrunden
2. De to potentionelle huller er rigtige huller, men baggrunden er monotom

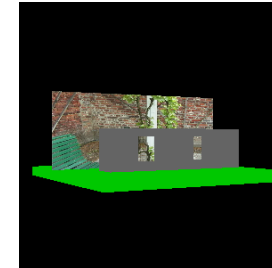
# Eksempel på et rigtigt hul



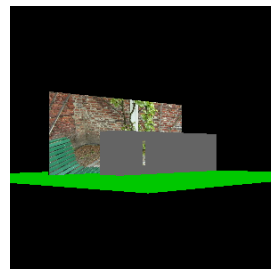
Ønsket view



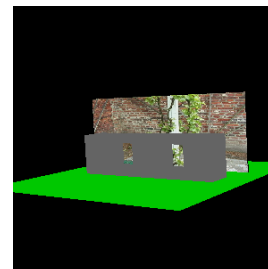
Reference view 1



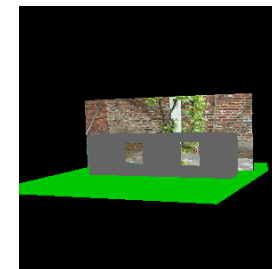
Reference view 2



Reference view 3



Reference view 4



Reference view 5

- De to potentionelle huller bestemmes til rigtige huller, da flere views observerer forskellige farvefordeling
- Reference view 3 benyttes ikke til bestemmelse af den potentionelle hul til højre, da der ikke er hul i silhouetten

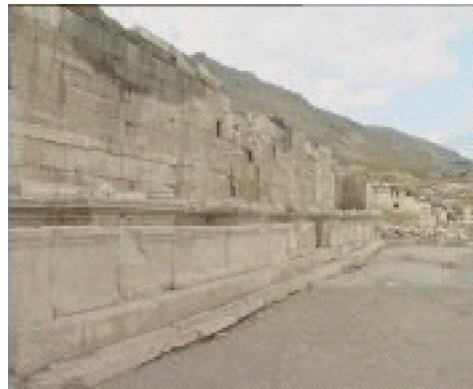


# Visualisering



# Visualisering

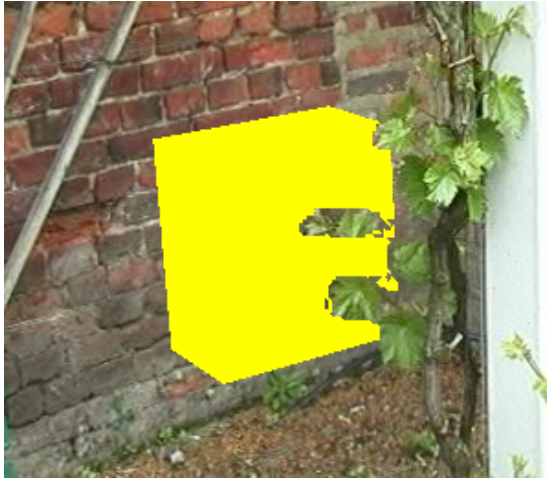
Arkæologisk udgravning ved Sagalassos i Tyrkiet



To typer af visualisering

- Rå billeder optaget af et videokamera
  - + Høj billedekvalitet, er tilgængelig
  - Jitter mellem virtuelle objekter og scenen
- Renderet billeder af 3D modellen
  - + Ingen jitter mellem virtuelle objekter og scenen
  - Ingen bevægelige objekter, samt lav billed kvalitet

# Visualisering



Ingen blur



5x5 Pixel Gaussian



9x9 Pixel Gaussian

- Blur med 5x5 Pixel Gaussian kerne giver det bedste visuelle resultat



# Forbedringer





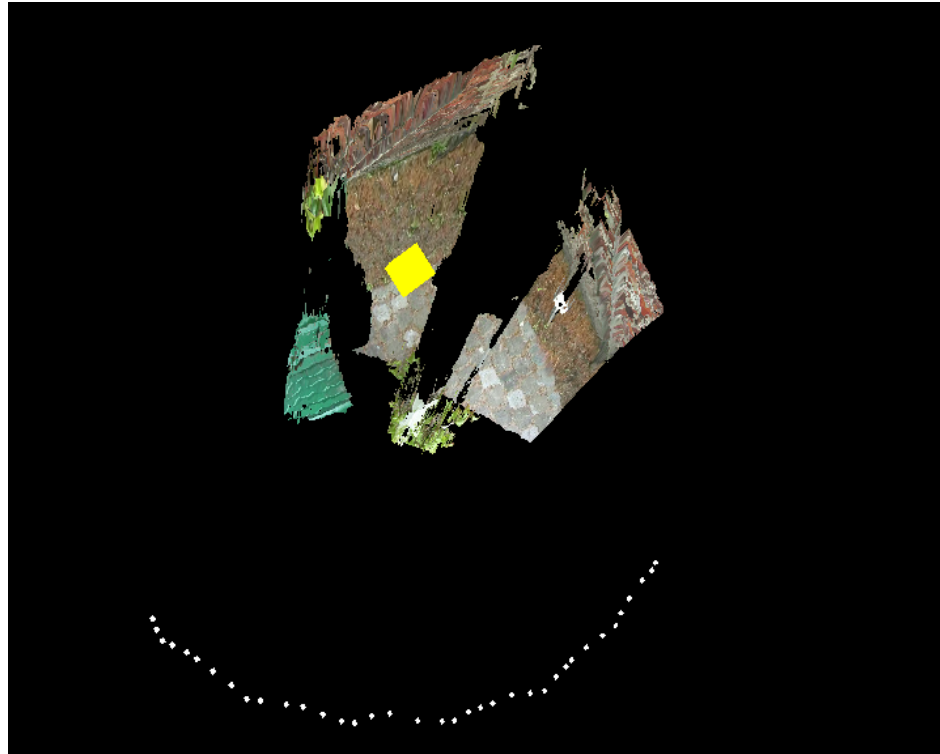
# Forbedringer

De mest betydende forbedringer, der vil øge rustheden af systemet

- Volumetric model af scenen
- Valg af gode views som indholder minimum af camouflage



# Valg af gode views



- Farveforskel mellem forgrundsobjekterne og baggrunden
- Ingen objekter mellem det valgte view og det volumen som udspændes af det ønskede view og de virtuelle objekter
- Stor vinkel mellem de valgte views

