

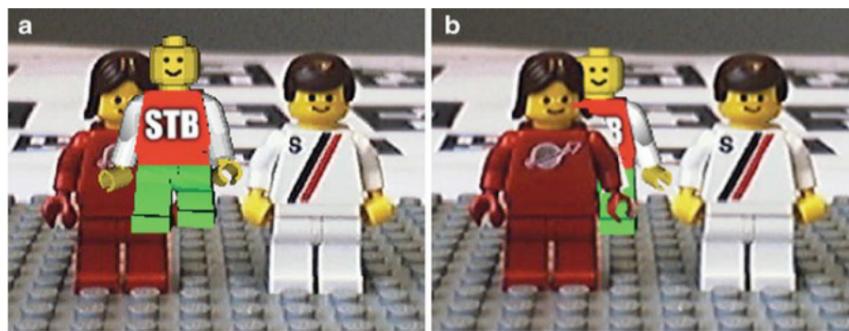
Nel caso dell'OST (Optical See Through) c'è un problema aggiuntivo, dato dal fatto che l'utente vede gli oggetti reali a distanza reale, mentre per vedere gli oggetti virtuali deve guardare la lente dove questi sono proiettati.

Normalmente le persone sono in grado di adattarsi senza troppe difficoltà, ma questo meccanismo può portare a stanchezza e malessere a lungo andare.

## Occlusione e phantom

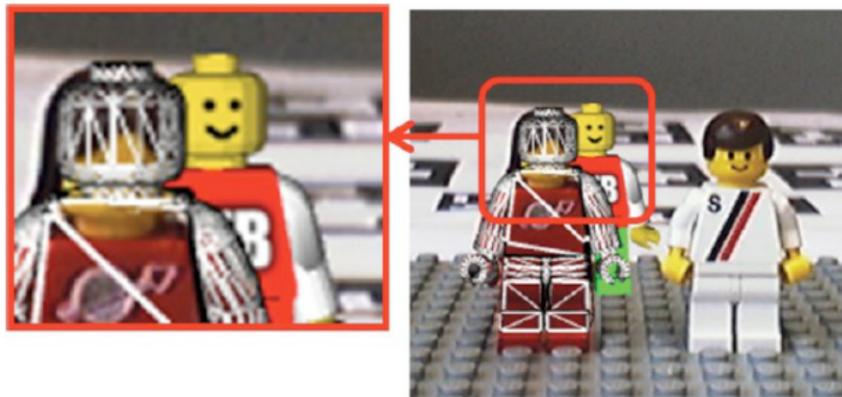
L'occlusione tra oggetti virtuale può essere raggiunta con un "z-buffer", ovvero un buffer di profondità. Questo è un tipo di dato che rappresenta la profondità degli oggetti 3D. Questa tecnica può essere usata quando abbiamo informazioni sul mondo reale. Quindi se ho una rappresentazione geometrica, digitale, del mondo reale, allora posso identificare gli oggetti del mondo reale, capire se stanno occludendo o meno qualcosa.

Per esempio nell'immagine b l'omino lego virtuale viene aggiunto alla profondità corretta, occluso dagli oggetti del mondo reale.



Nell'AR, il caso in cui l'oggetto virtuale è di fronte è più facile da gestire perché automaticamente andrà ad occludere gli oggetti del mondo reale, mentre invece se l'oggetto virtuale deve essere dietro un oggetto del mondo reale allora l'occlusione sarà più complicata da ottenere.

Un "phantom" è la rappresentazione virtuale di un oggetto reale, che viene renderizzato invisibilmente, ovvero solo il z-buffer viene modificato. Questo stabilisce i corretti valori di profondità per l'oggetto reale, di modo che quello virtuale possa essere mostrato con l'occlusione corretta.



Quindi disegnerò solo la parte dell'omino che sarà visibile.

Se non avessi il modello del mondo reale, questo sarebbe difficile fare, perché non sarei in grado di creare il phantom.

I problemi principali di questa tecnica sono:

- I modelli virtuali non rappresentano alla perfezione quelli reali
- Se il tracking del mondo reale e quello virtuale non è corretto, la parte occlusa sarà fuori posizione

Nuove tecnologie permettono di operare in ambienti dinamici in **real time**, dove non è possibile avere i phantoms. La depth map viene quindi ottenuta dinamicamente tramite hardware come sensori di profondità.

A volte vado anche a giocare con le **luci** per avere una migliore occlusione. Per esempio se devo occludere un oggetto reale con uno virtuale, ma quello reale ha un contrasto e una luminosità molto alta, rischio di non riuscire a nasconderlo. Di conseguenza anche l'oggetto virtuale deve essere proiettato con una luminosità molto alta.

## Risoluzione e refresh rate

Il **refresh rate** è la frequenza a cui l'immagine viene aggiornata sullo schermo, deve essere di almeno 60Hz.

La **risoluzione** è la capacità di mostrare un'immagine dettagliata. Nei sistemi VST è limitata, oltre dalla qualità del display, dalla qualità della telecamera.

La risoluzione e l'FOV hanno una relazione stretta, dato che un FOV più grande necessiterà di più pixel per mantenere la stessa densità.

In AR distinguiamo il **FOV dell'overlay** (la porzione di campo visivo occupata dalla visuale del visore AR, dove vengono mostrati gli elementi virtuali) e il **FOV periferico** (la restante parte del nostro campo visivo umano che non è coperta dal visore AR)

## Luminosità e contrasto

In situazioni all'aperto con tanta luce, la maggior parte dei display non sono in grado di avere abbastanza luce e contrasto per mostrare l'immagine.

Gli optical see through dipendono dalla trasparenza dello specchio combinatore, mentre i video see through possono cambiare la luminosità e il contrasto arbitrariamente.

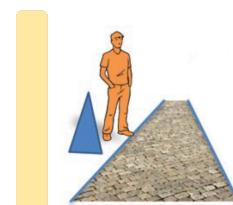
## Depth cues

Le depth cues possono essere **monoculari** (non hanno bisogno dell'uso di entrambi gli occhi) o **binoculari** (hanno bisogno di due occhi o un'immagine stereoscopica).

Le informazioni di profondità possono essere **assolute** (posizione nello spazio) o **relative** (posizione rispetto ad un altro oggetto).

Ci sono le **depth cues dinamiche** ovvero dove grazie al movimento, si nota la profondità. Anche il suono trasmette un'informazione di profondità.

La **prospettiva lineare** è un'indicazione di profondità basata sulla distorsione (esempio della strada che si stringe man mano che è più lontana).



**Relative size** diminuisce la grandezza degli oggetti, il più lontano **vengono proiettati** (esempio la texture delle mattonelle qua a destra che deve diventare più piccola man mano che si allontana).

**Grandezza conosciuta**, se c'è un oggetto (una persona per esempio) di cui conosciamo le dimensioni, questo aiuta a capire le dimensioni degli altri oggetti.

**Prospettiva atmosferica**, gli oggetti lontani non hanno troppi dettagli e troppo contrasto e hanno un colore verso il blu.



Fig 1

**Ombre** aiutano a capire la posizione e la dimensione degli oggetti, vedendo la distanza dell'oggetto dall'ombra.

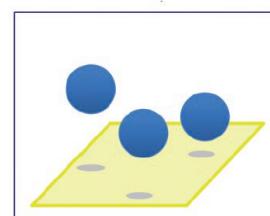


Fig 2

## Frame Cancellation

Se un oggetto virtuale raggiunge il bordo del display, l'illusione che l'oggetto sia di fronte al display viene a mancare.

## Esercizio



Strada: relative size + prospettiva lineare  
Lampione con persona a sinistra: known dimension  
Coppia con l'ombrellino e lampione: relative size, occlusion, known dimension  
Palazzi sfocati sullo sfondo: prospettiva atmosferica, prospettiva lineare  
Persone sulla strada, vicine e lontane: prospettiva lineare, known dimension

## Coppia a sinistra e la carrozza: occlusion

Selezionate le frasi che ritenete corrette riguardo i tipi di HMD

- Tutti gli HMD possono essere usati sia per VR che AR potrebbe non essere see-through 3% 1
- Un HMD Video See through permette di fare sia AR che VR 77% 27
- Qualunque HMD Optical See through permette di fare VR 3% 1
- Un Optical See through HMD può soffrire di problematiche di contrasto e luminosità 94% 33
- Un HMD Non See through permette di fare AR 6% 2

Selezionate le frasi che ritenete corrette riguardo all'occlusion

- L'occlusion è una delle principali informazioni sulla posizione relativa di un oggetto 100% 1
- In AR, l'occlusion si riferisce solo ad un oggetto virtuale che nasconde uno reale 0% 0
- Un phantom è una rappresentazione virtuale di un oggetto reale 100% 1
- Per usare z buffer e phantom non devo necessariamente avere un modello dell'ambiente reale 0% 0