

Technologia mowy w grach komputerowych

Plan prezentacji:

- Rozpoznawanie mowy w kontekście gier komputerowych
- Różne zastosowania ASR (Automatic Speech Recognition), dla różnych gatunków gier
- Od strony inżynierskiej – warunki, sposób tworzenia
- Przyszłość – multikontroler



Rozpoznawanie mowy, a gry

- Potrzeba usprawnienia systemu sterowania w grach komputerowych
- Rozpoznawanie sygnałów emitowanych przez człowieka.
Krok pierwszy: rozpoznawanie mowy
- Użyteczność:
 - ruchy manualne->ruchy manualne, mowa->mowa
 - szybkość
 - uproszczenie/umożliwienie sterowania
- Obecne zastosowanie: projekty studenckie, niewielkie aplikacje rozrywkowe, narzędzia do samodzielnego programowania
- Brak działań na skalę komercyjną



Praktyczne zastosowanie ASR w zależności od rodzaju gry

- FPP (First-Person Perspective)



Praktyczne zastosowanie ASR w zależności od rodzaju gry

- RTS (Real Time Strategy)



Praktyczne zastosowanie ASR w zależności od rodzaju gry

- MMO (Massively Multiplayer Online) i inne o ogromnych interfejsach



Praktyczne zastosowanie ASR w zależności od rodzaju gry

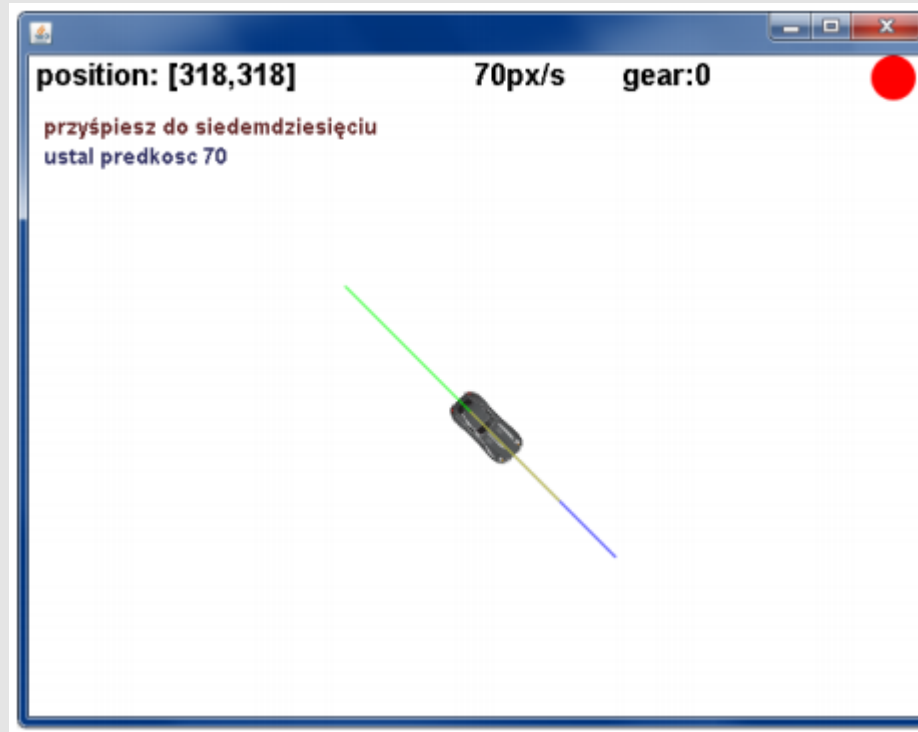
- Wyzwanie: cRPG (computer Role-Playing Game)



Praktyczne zastosowanie ASR w zależności od rodzaju gry

- Nowe rodzaje gier oparte na rozpoznawaniu mowy

Dariusz Wawer: „System rozpoznawania mowy jako interfejs gry komputerowej” - pilot rajdowy



Techniczny punkt widzenia

Bardzo skuteczny system rozpoznawania mowy (oparty na HMM)

- Rozpoznawanie bliskie 100%
- Różni mówcy (ew. personalizacja treningu), wykorzystujący różne mikrofony (ew. mikrofony dedykowane) – problem zakłóceń/jakości
- Możliwie duża odporność na wady wymowy, zbyt szybkie wypowiedanie słów itp.
- Maksymalna szybkość działania – dynamika gry
- Reakcja modelu językowego na wydarzenia w grze
- Projektowanie systemu otwartego na różne wersje językowe



Przyszłość

Multikontroler

- Sterowanie manualne + werbalne + BCI (Brain-Computer Interface)

Na zdjęciu: Emotiv Epoc (panel rozróżniający ponad 30 stanów mentalnych)



- Przykłady: Harry Potter, Mount&Blade

