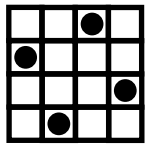
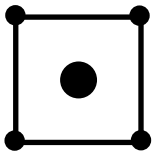
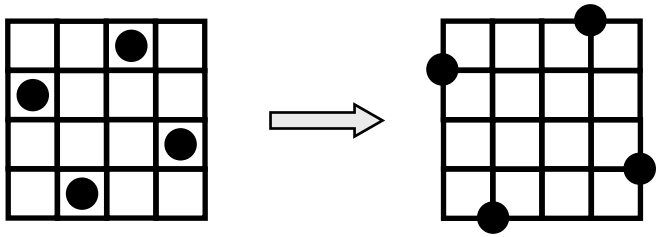
Das FLIPQUAD-Pattern

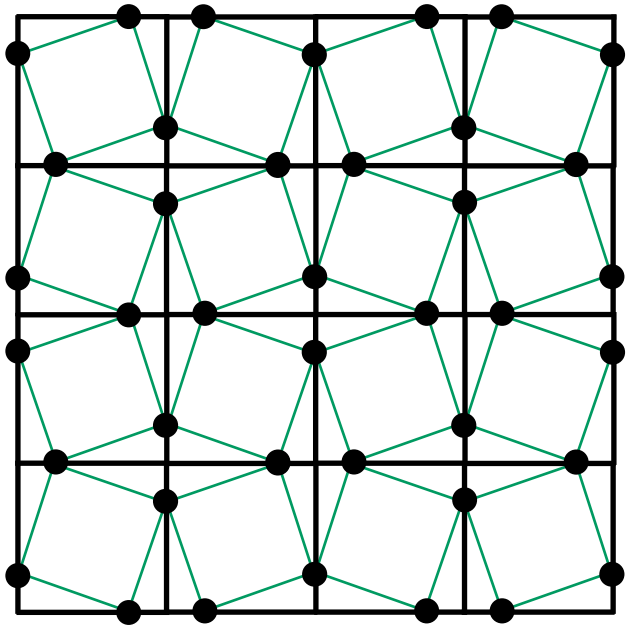
[Möller 2003]

- Vorteil von RGSS:
  - Ein Sample pro Spalte und Zeile
- Vorteil von Quincunx:
  - Sample-Sharing
- Kombiniere Vorteile von RGSS und Quincunx:





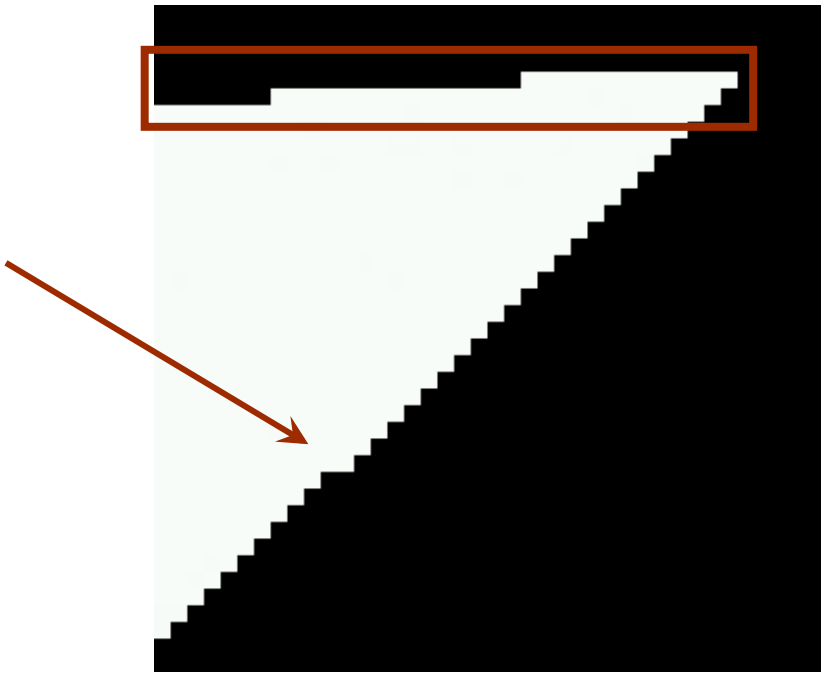




  - Gewichte: 0.25 pro Sample
  - Kosten: 2 Samples / Pixel
  - Qualität: wie RGSS

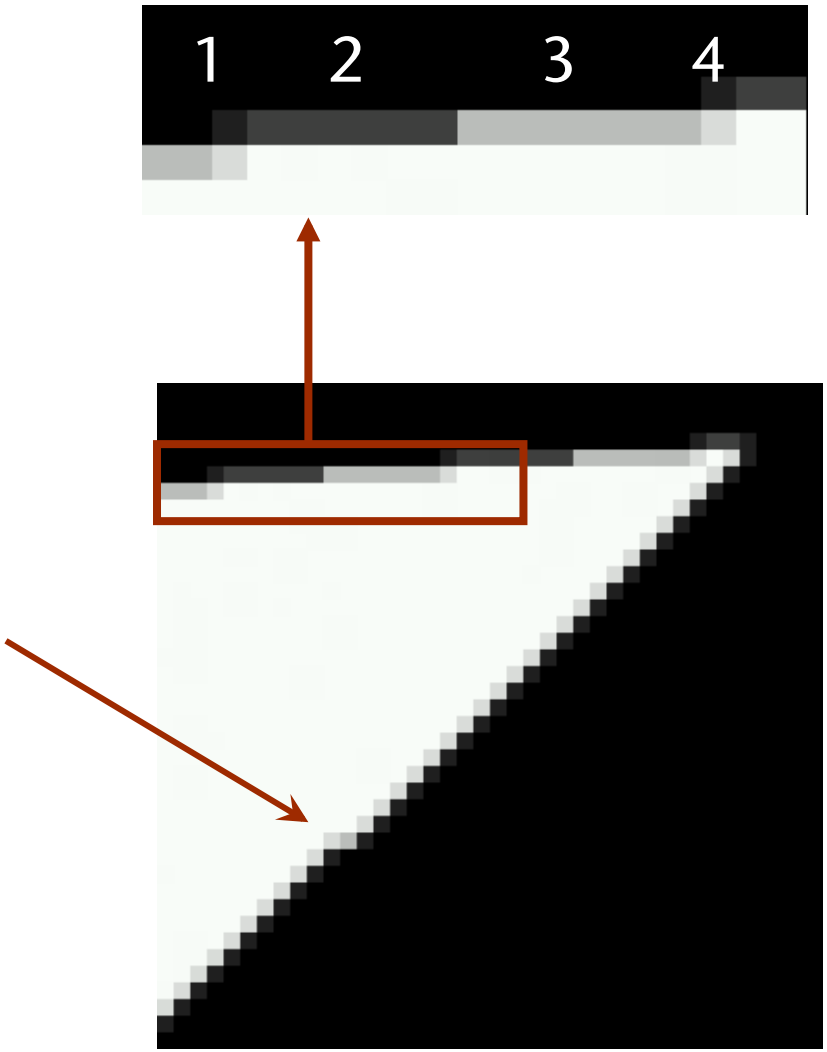
Vergleich von FLIPQUAD mit Quincunx

- Zunächst ohne Anti-Aliasing:
  - Aliasing an annähernd horizontal verlaufenden Kanten
  - Sprünge wenn Steigung annähernd 45°

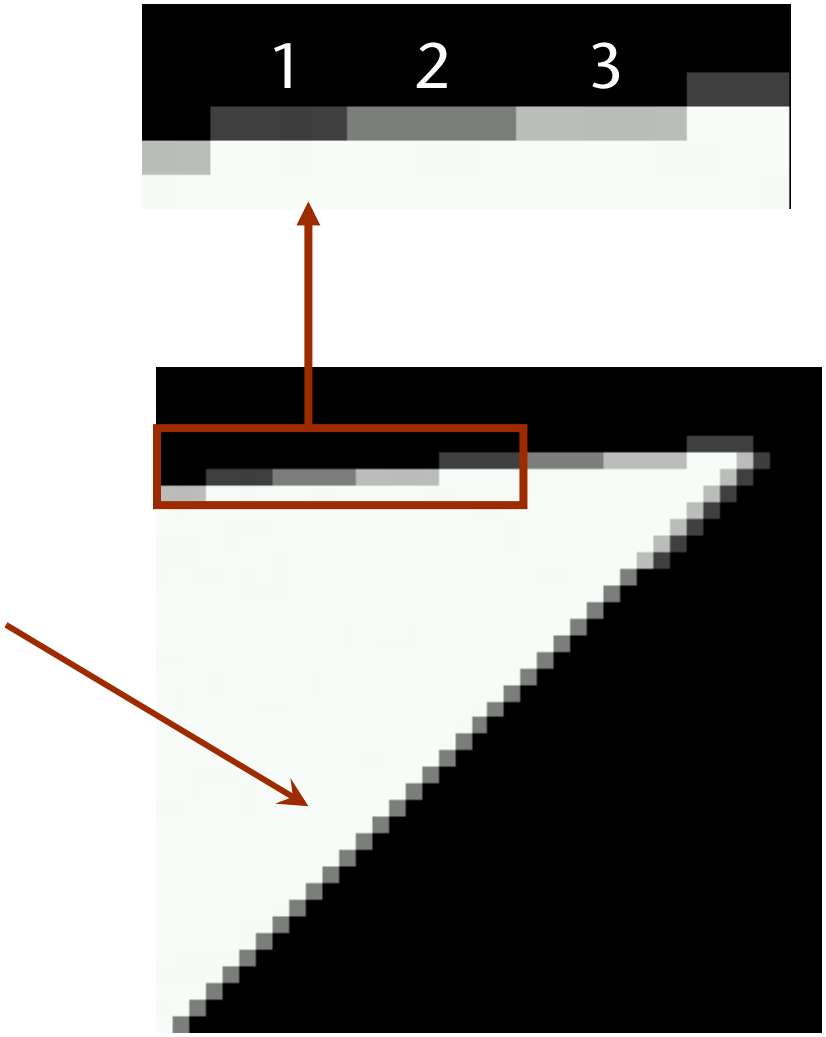


1 Sample / Pixel

- Quincunx:
  - 4 Grauwerte, aber 2 davon kommen bei annähernd horizontalen Kanten (fast) nicht vor
  - Sprünge bei einem Winkel von annähernd 45° sind immer noch sichtbar



- FLIPQUAD:
  - Nur 3 Grauwerte,
    aber gleichmäßig verteilt
  - Keine Sprünge



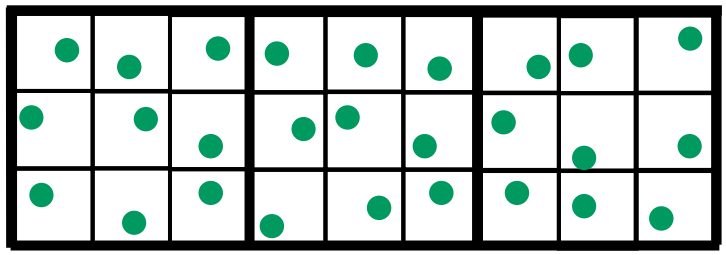
- Beispiel einer vollständigen Szene

1 Sample Quincunx FLIPQUAD

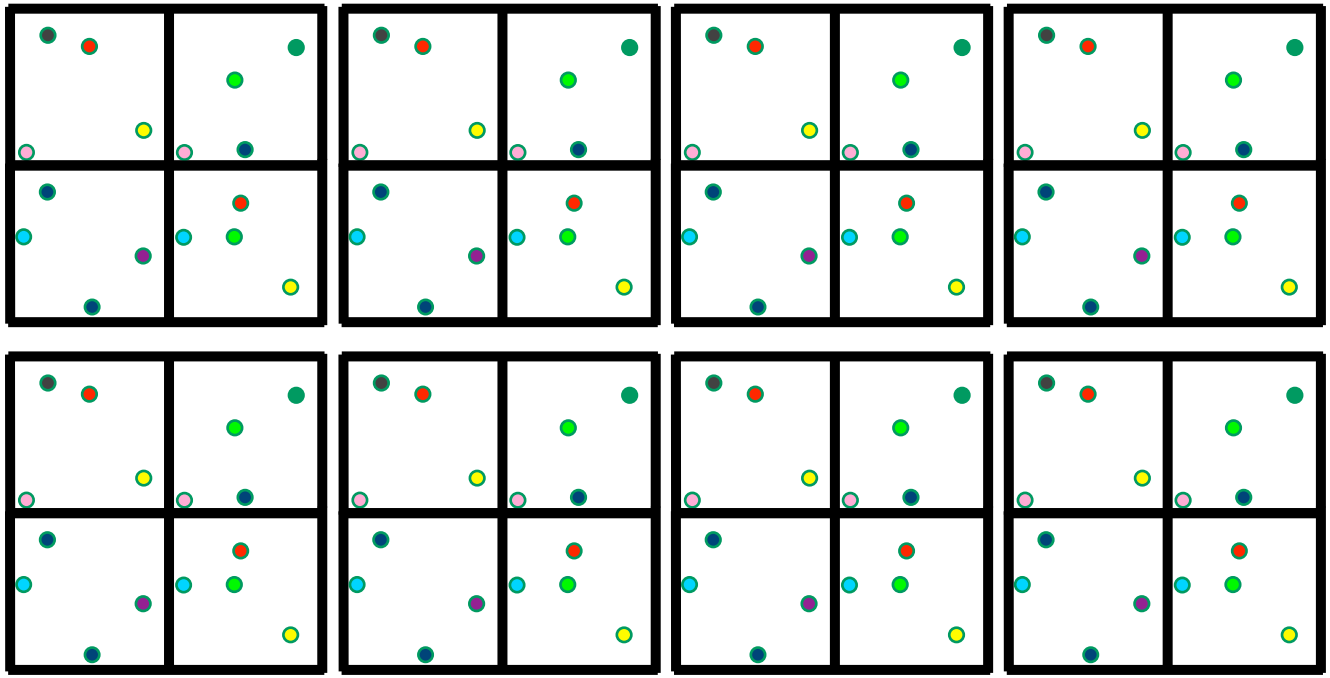
- Finnische Forschungsgruppe hat gezeigt, das FLIPQUAD das beste Samplingverfahren bei 2 Sample/Pixel ist
- FLIPQUAD ist in ATI/Bitboy's Architektur implementiert

Stochastisches Antialiasing (jittered sampling)

- Reguläres Sampling kann Aliasing nicht eliminieren – nur reduzieren!
- Weil es zu jedem regulären Pattern eine (oder mehrere) Steigungen gibt, so daß Kanten mit dieser Steigung deutlich weniger verschiedene Intensitätsstufen produzieren.
- Jittering ersetz Aliasing durch Rauschen
- Achtung: völlig zufällige Samples können "schlecht" liegen
- Lösung: eine Art stratifiziertes Sampling
  - Unterteile Pixel in n x n Subpixel, wähle innerhalb des Subpixel die Position zufällig
- Beispiel:



- Sample Position dürfen räumlich abweichen aber nicht zeitlich
  - Jedes Pixel muss in allen Frames die Samples gleich anordnen



  - Image von ATIs SMOOTHVISION Sampling
  - Muster ist pseudo-Zufall und wiederholt sich → handlich für Hardware

Centroid-Interpolation

- Einige Pixel werden nur teilweise bedeckt
- Pixel Shader läuft einmal pro Pixel
- Zu interpolierende Größe wird für Pixel-Mittelpunkt interpoliert (vomRasterizer):
  - Pixelmittelpunkt kann aber außerhalb des Dreiecks liegen
  - Efektiv findet dann Extrapolation statt!
  - Abhängig von der Bedeutung der Interpolation kann dies sehr schlecht sein
- Lösung: *Centroid-Interpolation* wertet Interpolationsgröße am Mittelwert (Centroid) der überdeckten Samples aus: