

# The MorphoxX Project

*Jean-Baptiste Fiot*

March 2009



© SEGA



# Goal and motivations

- Goal: tool to deform quickly 3D model
  - Cage-based methods
    - 3 coordinate systems studied in this project
- **Let's do a quick demo!**
- Applications:
  - Animated movies
  - Video games



© SEGA



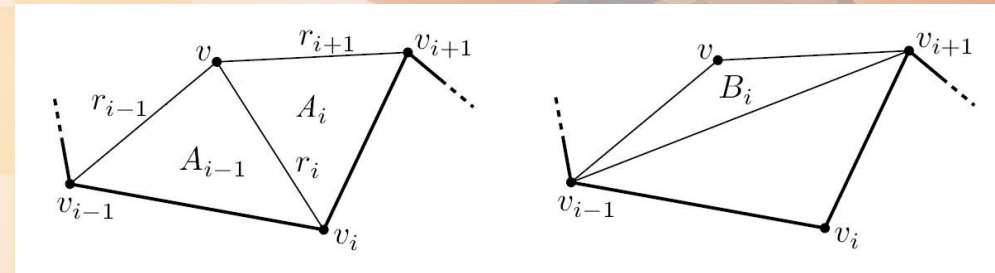


# Mean Value Coordinates

- Generalized barycentric coordinates.

$\{w_i : \mathbb{R}^2 \rightarrow \mathbb{R}, i \in [1, n]\}$  so that  $\sum_{i=1}^n w_i(v)(v_i - v) = 0$

$$w_i = \frac{r_{i-1} A_i - r_i B_i + r_{i+1} A_{i-1}}{A_{i-1} A_i}$$



- Coordinates are normalized in the end

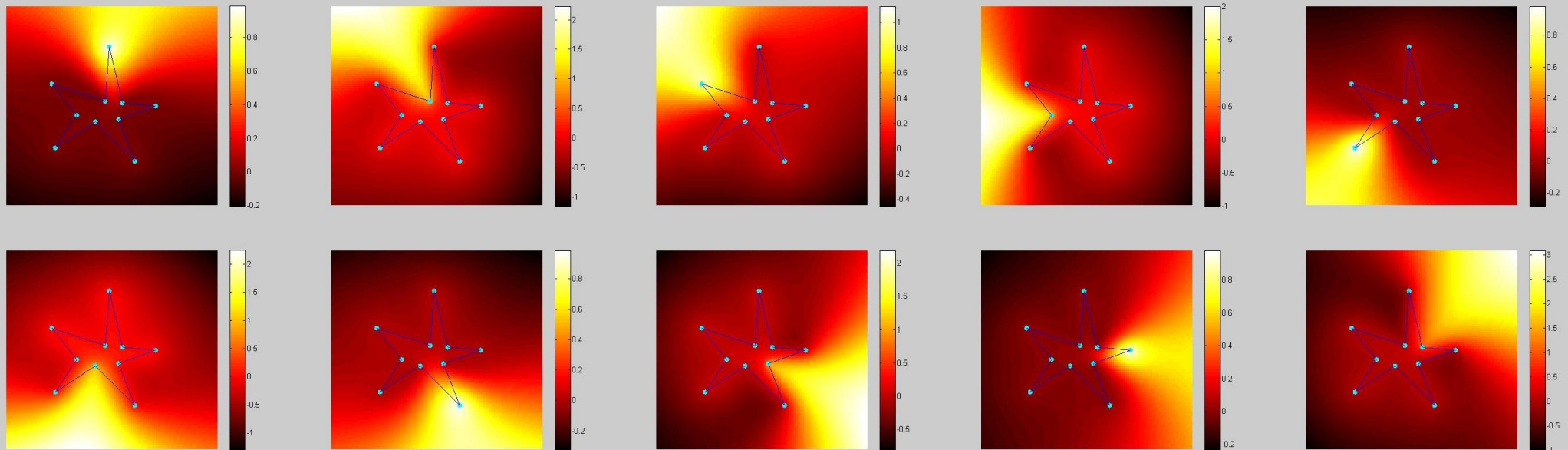
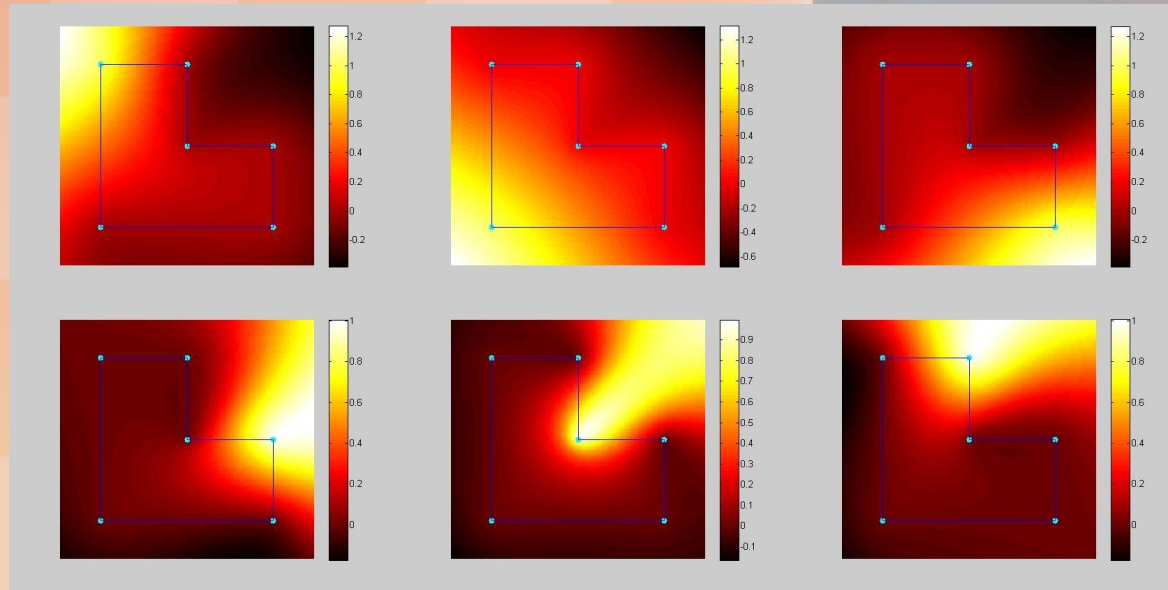


© SEGA

SONIC TEAM

SEGA

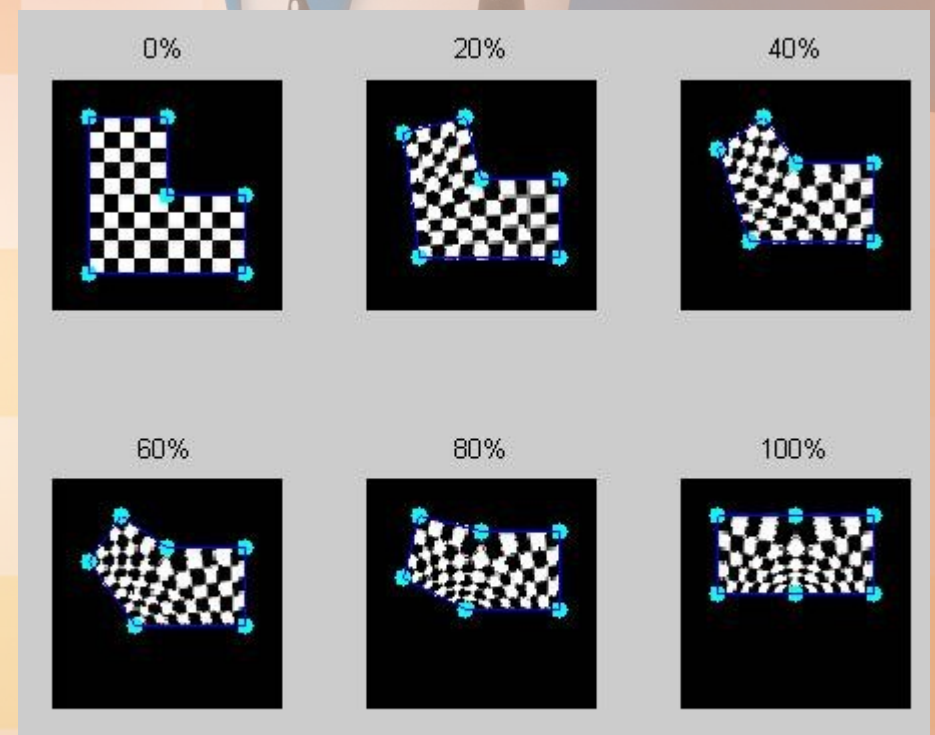
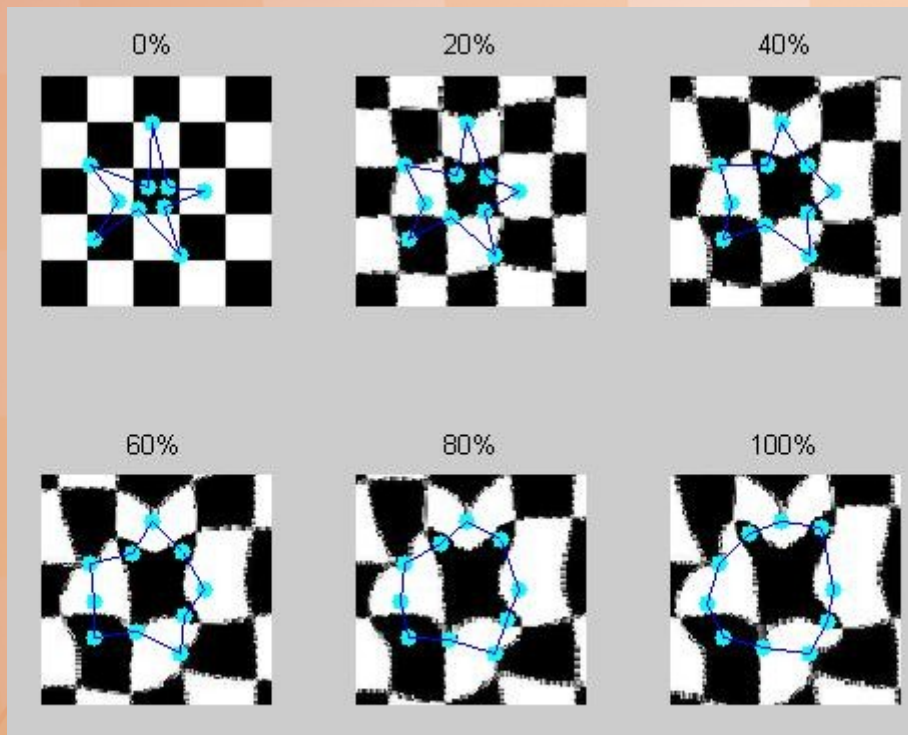
# Mean Value Coordinates





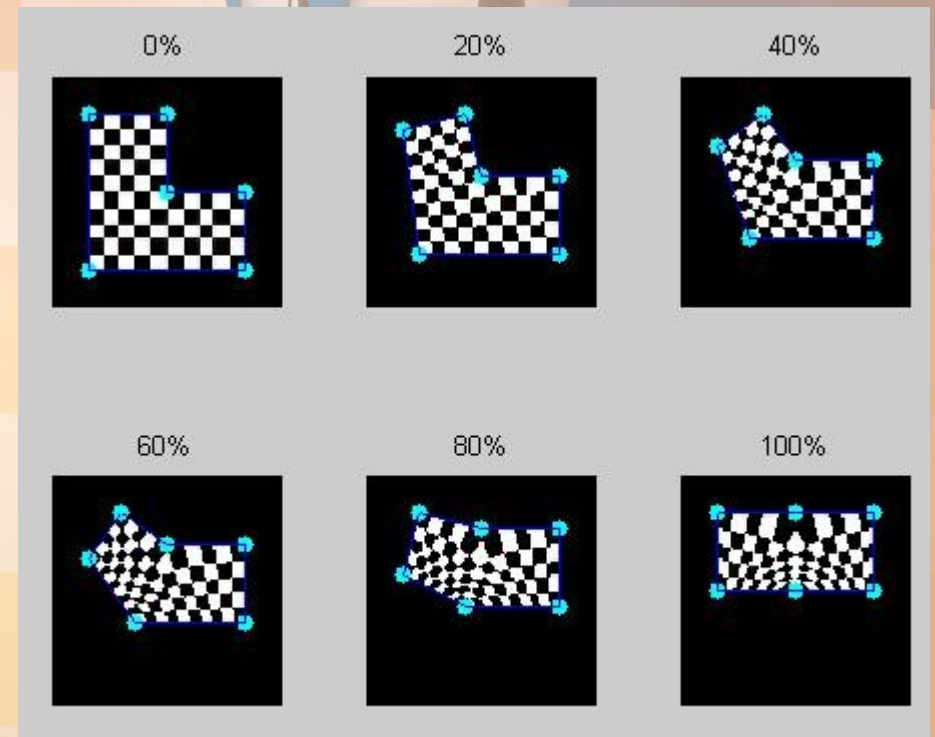
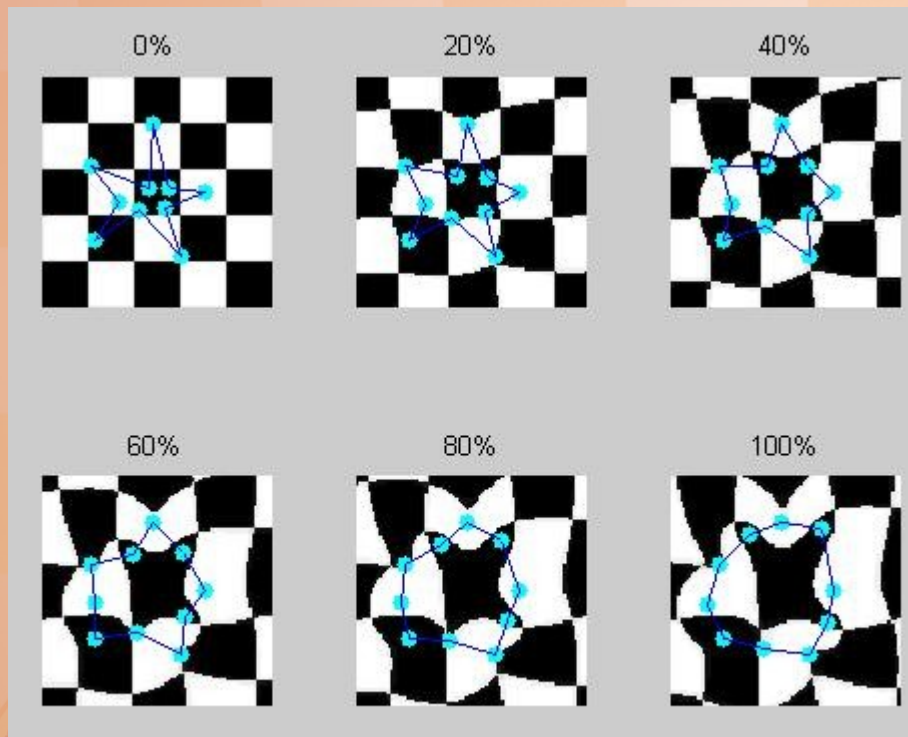
# Mean Value Coordinates

- Progressive deformations:



# Mean Value Coordinates

- Using nearest pixel instead of interpolation:





# Mean Value Coordinates



© SEGA

SONIC TEAM

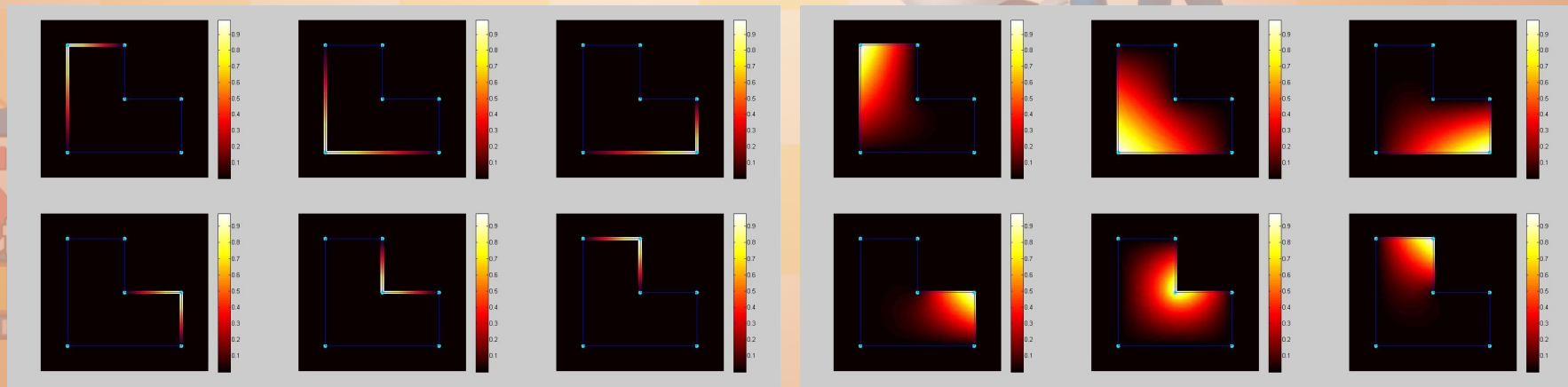
SEGA

# Harmonic Coordinates

- Defined on the interior of the cage
- $\forall i \in \llbracket 1, n \rrbracket, \forall v \in \mathring{C}, \Delta w_i(v) = 0$
- Creation of a mask
- Initialization on the boundaries
- Propagation in the interior via 4c-averaging



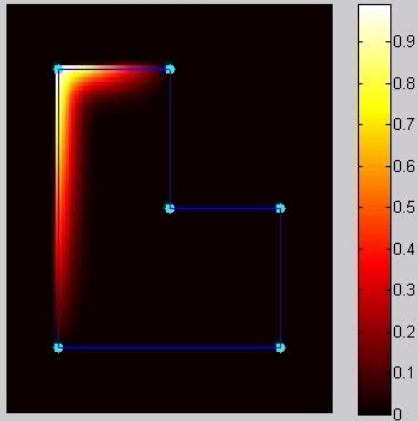
© SEGA



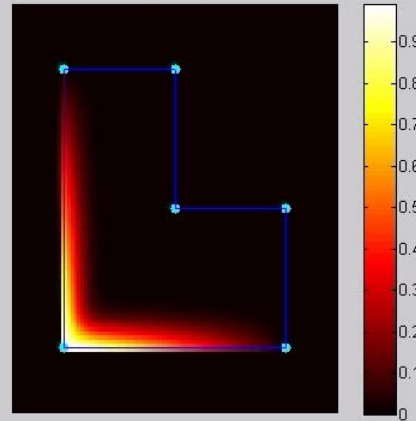


# Harmonic Coordinates

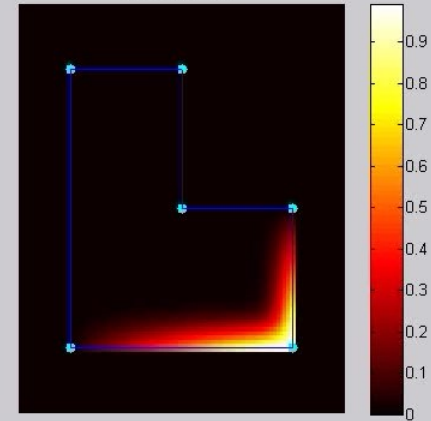
Vertex 1 Iter #46



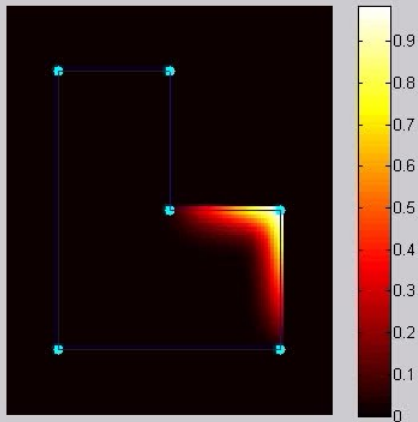
Vertex 2 Iter #46



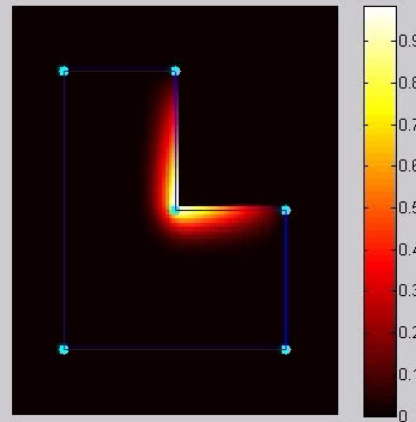
Vertex 3 Iter #46



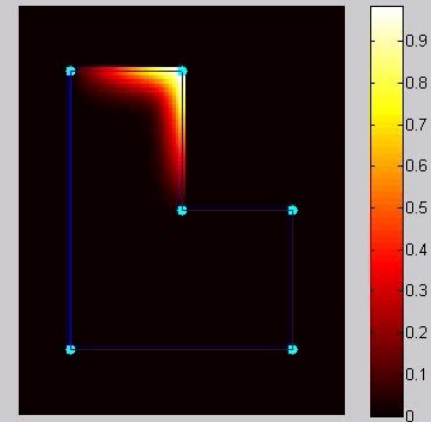
Vertex 4 Iter #46



Vertex 5 Iter #46

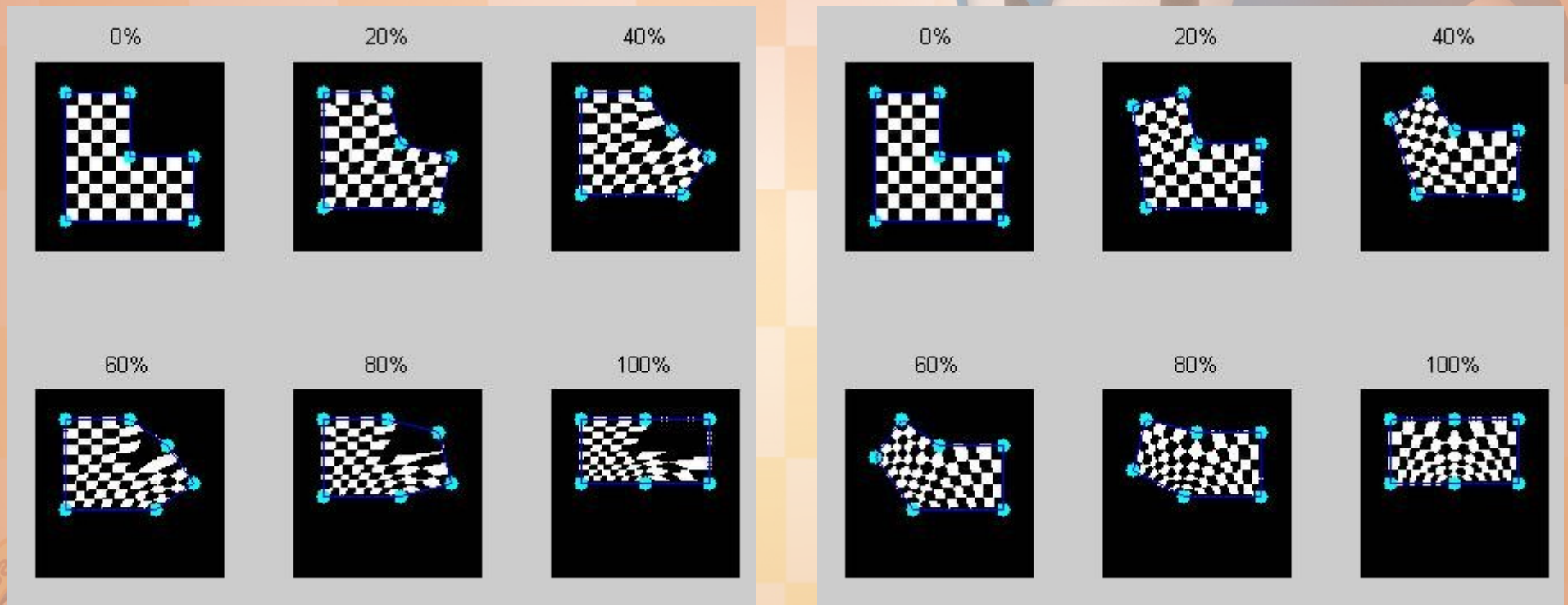


Vertex 6 Iter #46



# Harmonic Coordinates

- Progressive deformations:



© SEGA

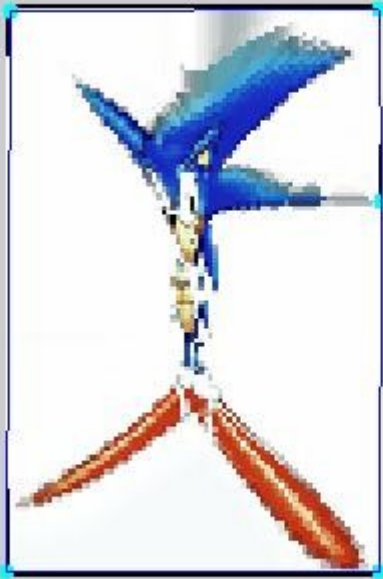
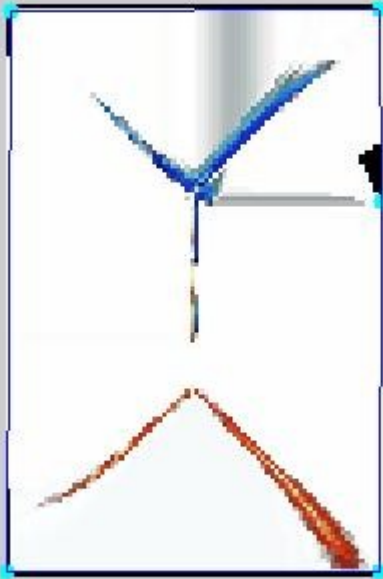
SONIC TEAM

SEGA



# Harmonic Coordinates

- Require enough iterations
  - Examples with 100, 500, 2000, 5000, and 20000 iterations:



SONIC  
THE HEDGEHOG

© SEGA

SONIC  
TEAM

SEGA

# Harmonic Coordinates

- Sonic 300\*450, 29 vertices, 25000 iterations
  - Born after 21 hours of computation on a Dell XPS Gen 2 with proc Intel Pentium M 2.13 GHz... but he is cute, isn't he?





# Green Coordinates

- Added terms based on face elements to have coordinates respecting the cage's faces' orientations

- $$v = \sum_{\text{vertices}} \phi_i(v) v_i + \sum_{\text{edges}} \psi_i(v) n(t_i)$$

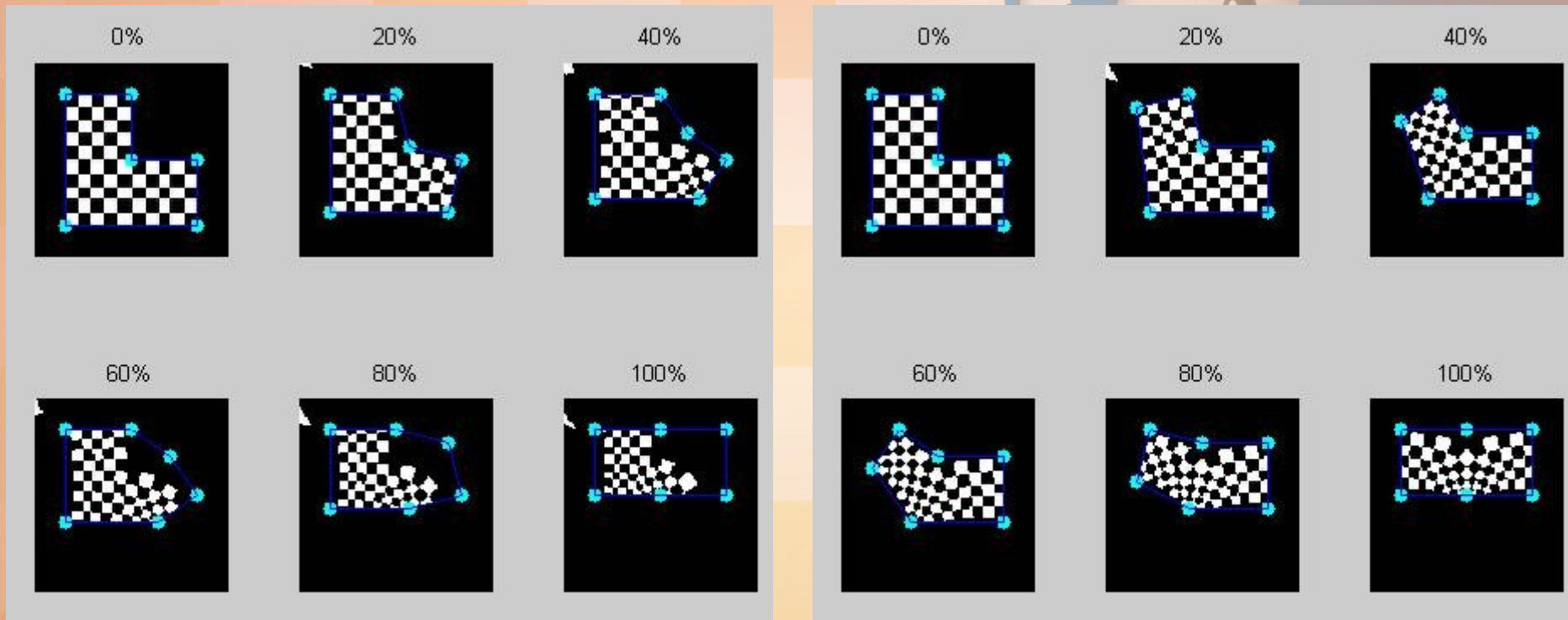
- $$v' = \sum_{\text{vertices}} \phi_i(v) v'_i + \sum_{\text{edges}} \psi_i(v) s_i n(t'_i) \quad \text{with } s_i = \frac{\|t'_i\|}{\|t_i\|}$$

- **TRICK:** Normalisation of the vertex-based terms (to be origin invariant: previous formulation: affine space  $\leftrightarrow$  vectorial space)



# Green Coordinates

- Progressive deformations:



© SEGA





# Green Coordinates

- Note the flexibility to allow the model to step out of the cage (continuity outside not implemented)



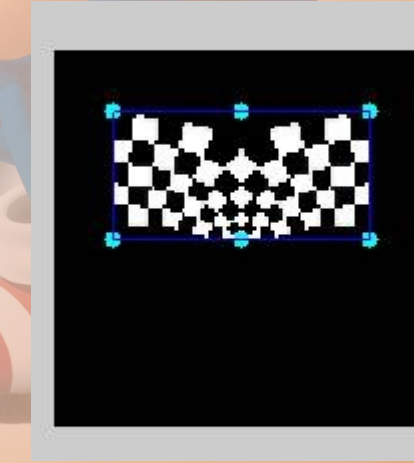
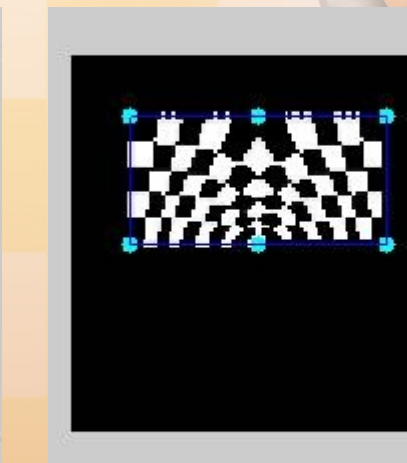
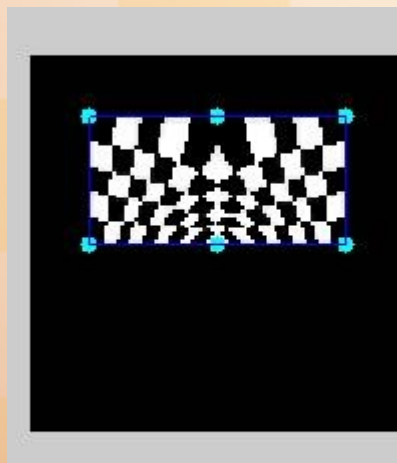
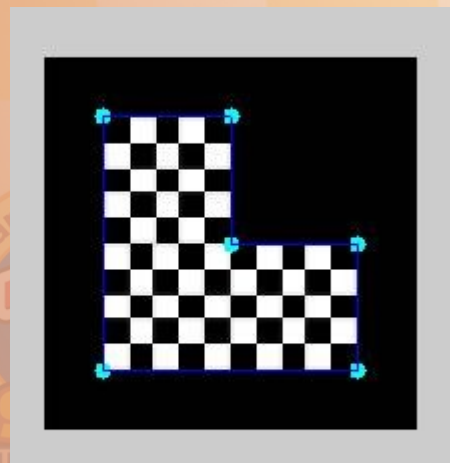
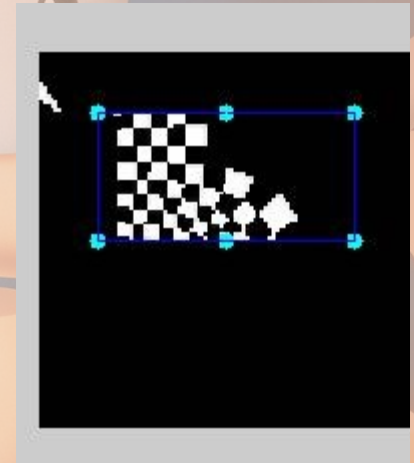
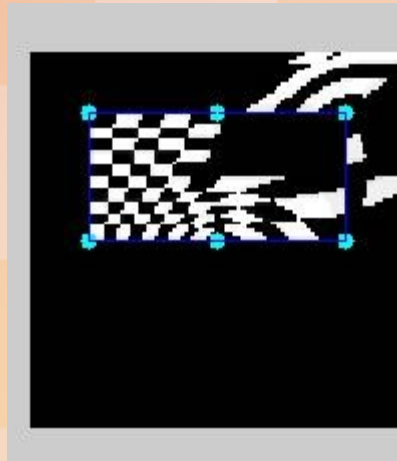
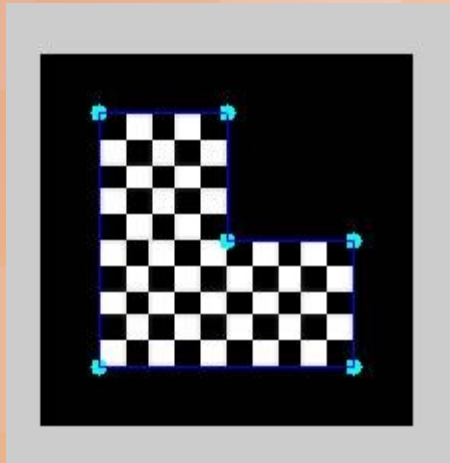
© SEGA

SONIC TEAM

SEGA

# Comparison

- Original – MVC – HC – GC:





# Comparison

- Initial – MVC – HC – GC:



© SEGA

SONIC TEAM

SEGA

# Conclusion

	MVC	HC	GC
Computing time	★ ★ ★	★	★ ★
Simplicity	★ ★ ★	★ ★ ★	★
Flexibility	No	No	Yes
Defined everywhere	Yes	No	Yes
Partial deformations	★	No	★ ★ ★

- Source code available at <http://code.google.com/p/morphoxx>



© SEGA

