Real-time Deep Dynamic Characters

MARC HABERMANN, LINGJIE LIU, Max Planck Institute for Informatics
WEIPENG XU, MICHAEL ZOLLHOEFER, Facebook Reality Labs
GERARD PONS-MOLL, AND CHRISTIAN THEOBALT, Max Planck Institute for Informatics

CCS Concepts: • **Computing methodologies** → **Motion capture**; *Motion capture*; *Mesh geometry models.*

Additional Key Words and Phrases: human modeling, human performance capture, deep learning, non-rigid surface tracking

ACM Reference Format:
Marc Habermann, Lingjie Liu, Weipeng Xu, Michael Zollhoefer, and Gerard Pons-Moll, and Christian Theobalt. 2020. Real-time Deep Dynamic Characters. *ACM Trans. Graph.* 40, 4, Article 94 (July 2020), 2 pages. https://doi.org/10.1145/3450626.3459749

In Tab. 1 and Tab. 2, we provide an overview of all symbols used in the main document and their respective descriptions.

Table 1. Symbols used in the main document and their descriptions.

Notations	
Symbol	**Description**
$\mathcal{T}_{\text{st}}$	Static texture map
s_i	Per-vertex rigidity weights [Habermann et al. 2019]
$\mathcal{S}$	Skeletal pose
$\boldsymbol{\theta}$	Skeletal joint angles
$\boldsymbol{\alpha}$	Skeletal root rotation
$\mathbf{z}$	Skeletal root translation
$\mathcal{G}$	Embedded graph
K	Number of embedded graph nodes
$\mathbf{A}$	Embedded graph rotation parameters
$\mathbf{T}$	Embedded graph translation parameters
$\mathbf{a}_k$	Embedded graph rotation parameters of node k
$\mathbf{t}_k$	Embedded graph translation parameters of node k
$\mathcal{N}_{\text{n}}(k)$	Connectivity of the graph node k
$w_{i,k}$	Vertex-to-node weights
$\mathcal{N}_{\text{vn}}(i)$	Set of nodes that influence vertex i
$\mathbf{D}$	Per-vertex Displacements
$\mathbf{d}_i$	Displacement of vertex i
C_i	Character deformation of vertex i
$\hat{\mathbf{v}}_i$	Undeformed template vertex i
$\mathbf{g}_k$	Position of the undeformed graph node k
R	Rotation angles to rotation matrix function
$R_{\text{sk},k}$	Dual quaternion to rotation matrix function
$t_{\text{sk},k}$	Dual quaternion to translation vector function
C	Number of cameras
$\mathcal{I}_{c,f}$	Frame f of camera c
$\mathcal{D}_{c,f}$	Distance transform image of frame f of camera c
$\mathcal{F}_{c,f}$	Foreground mask of frame f of camera c
$\hat{\mathcal{F}}_{c,\mathbf{u}}$	Eroded foreground mask of frame f of camera c
$\mathcal{S}_f$	Tracked motion of frame f
$\mathcal{M}_t$	Motion window for frame f
$\hat{\mathbf{z}}_t$	Normalized root translation for frame t
$\hat{\boldsymbol{\alpha}}_{y,t'}$	Normalized joint angle for frame t
$\hat{\mathcal{M}}_t$	Normalized motion window for frame t
$\hat{\mathcal{S}}_f$	Normalized skeletal pose for frame t
$\hat{\mathcal{M}}_t$	Normalized motion window for frame t
F	Window size
$\hat{\mathcal{M}}_{\text{eg}}$	Normalized motion in graph representation
f_{eg}	EGNet
$\mathbf{w}_{\text{eg}}$	EGNet learnable weights
$\mathbf{w}_{\text{delta}}$	DeltaNet learnable weights

Authors' addresses: Marc Habermann, Lingjie Liu, Max Planck Institute for Informatics, Campus E1, Stuhlsatzenhausweg 4, Saarbruecken, Saarland, Germany, 66123, mhaberma@mpi-inf.mpg.de, lliu@mpi-inf.mpg.de; Weipeng Xu, Michael Zollhoefer, Facebook Reality Labs, District Fifteen: 131 15th Street, Pittsburgh, Pennsylvania, USA, 15222, wxu@mpi-inf.mpg.de, zollhoefer@fb.de; Gerard Pons-Moll, and Christian Theobalt, Max Planck Institute for Informatics, Campus E1, Stuhlsatzenhausweg 4, Saarbruecken, Saarland, Germany, 66123, gpons@mpi-inf.mpg.de, theobalt@mpi-inf.mpg.de.

0730-0301/2020/7-ART94
https://doi.org/10.1145/3450626.3459749

Table 2. Symbols used in the main document and their descriptions.

Notations	
Symbol	**Description**
$u_{k,l}$	Per-node connection rigidity weights
$\Phi_{c,\mathbf{u}}$	Rendering function for camera c and pixel $\mathbf{u}$
$a_{c,\mathbf{u}}$	Albedo color of camera c and pixel $\mathbf{u}$
$i_{c,\mathbf{u}}$	Illumination of camera c and pixel $\mathbf{u}$
$v_{c,\mathbf{u}}$	Visibility of camera c and pixel $\mathbf{u}$
$t_{c,\mathbf{u}}$	Barycentric coordinates of camera c and pixel $\mathbf{u}$
$n_{c,\mathbf{u}}$	Surface normal of camera c and pixel $\mathbf{u}$
$\mathbf{l}_{c,j}$	Lighting coefficients
$\mathbf{I}_{\text{sh}}$	Identity lighting
$\mathbf{l}^{*}_{\text{mcs},c}$	Optimized lighting coefficients
g	RGB to YUV color transform
$\mathcal{I}_c$	Frame of camera c
$\mathcal{N}_i$	Set of neighbouring template vertices of vertex i
$\mathcal{T}_{\text{norm}}$	Normal texture
$\mathcal{T}_{\text{cam}}$	Camera texture
$\mathcal{T}_{\text{dyn}}$	Dynamic texture
π_c	Camera projection of camera c
$\rho_{c,i}$	Normal matching function
$\mathcal{B}_c$	Set of boundary vertices for camera c
$\mathbf{V}_{\text{fi}}$	Densely deformed vertices
$\mathbf{v}_{\text{fi},i}$	Densely deformed vertex i
$\mathbf{V}_{\text{co}}$	Coarsely deformed vertices
$\mathbf{v}_{\text{co},i}$	Coarsely deformed vertex i

REFERENCES

Marc Habermann, Weipeng Xu, Michael Zollhoefer, Gerard Pons-Moll, and Christian Theobalt. 2019. LiveCap: Real-time Human Performance Capture from Monocular Video. *ACM Trans. Graph.* (2019).