

# Appendix - T2Bs: Text-to-Character Blendshapes via Video Generation

## 1. Expression prompts

We generate videos using expression prompts concatenated with the original character prompt and camera motion description. To ensure a diverse range of head motions for virtual characters, we predefine a set of expression prompts that encompass various potential movements. These expressions can be categorized into two groups:

**Physically specific expressions** – These describe concrete, observable actions involving facial features: talking, screaming, laughing, smiling, smirking, closing the mouth, opening the mouth extremely wide, blinking, teasing the eyes, looking around, waving the ears, tongue sticking out the mouth, shaking the head.

**Emotionally expressive states** – These convey the character’s inner feelings and overall demeanor: sad, angry, chilling, happy, pensive, confused, disappointed.

We observe that, on one hand, the character may exhibit additional expressions beyond those specified in the input prompt, such as closing the mouth while blinking. On the other hand, the video model may fail to generate the described expression as prompted, especially the emotionally expressive states. In the latter case, we retain the video if it still presents natural-looking motion.

## 2. T2Bs model Expressiveness Evaluation details

As demonstrated in Fig. 5 of the main paper, we fit T2B models to random captures that fall outside the model’s training range. Specifically, we fit the corresponding models on 10 identities, each with 5 held-out expression videos, and then fit the model to each frame of these videos. The average pixel-wise L2 fitting error is 0.0009, while the average 3D point-to-point error is 0.0017 relative to the bounding box size.

## 3. Analysis on the Number of Control Points

To ensure scalability, we use pre-defined control points from the static asset instead of jointly optimizing them across all expression videos. Specifically, we first apply

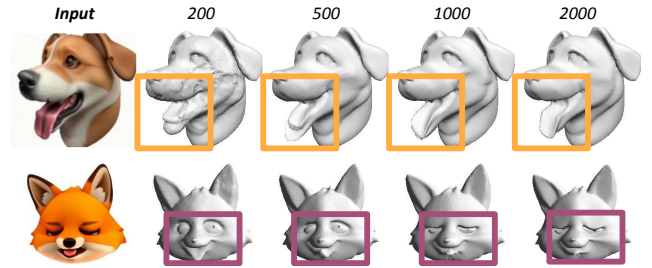


Figure 1. Ablation study on the number of control points. Using 2000 joints captures fine-grained motions, such as tongue (1st row) and eyelid (2nd row) movements, better than 200, 500, or 1000 joints.

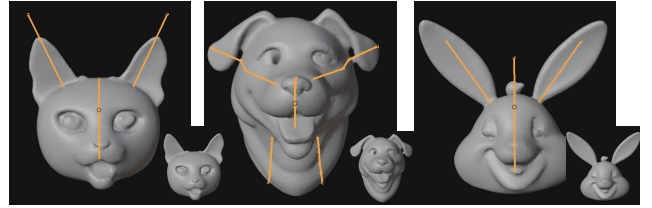












Figure 2. (Skeleton predicted by UniRig (orange), which isn't suitable for facial motion.

the KNN algorithm to select  $k = 2000$  uniformly distributed control points. For each Gaussian, we assume it is influenced by its 10 nearest control points. The blending weights for these neighbors are computed as the normalized inverse Mahalanobis distances. We then fix both the Gaussian-to-control-point associations and the blending weights, and optimize only the transformation of each control point. This allowing new expression videos to be incorporated without requiring re-optimization of previously processed videos. Also, this modular approach avoids the need for computationally expensive joint optimization across an ever-growing dataset. We show parameter analysis on the the number of control points in Fig 1. 2000 control points captures fine-grained motions, such as tongue (1st row) and eyelid (2nd row) movements, better than 200, 500, or 1000 joints.

Beyond KNN control points, we also try to obtain skeletons predicted by, MagicArticulate, and UniRig. We show a few examples of bones and control points prediction in Fig. 2, which is not suitable to model facial expressions.

image quality	LPIPS↓	FID↓	geometry	p2p↓	NC↓	geometry	p2p↓	NC↓
SV4D	0.1543	167.3	T2Bs	<b>0.0576</b>	<b>0.1611</b>	T2Bs	<b>0.0476</b>	<b>0.1671</b>
4Real-Video	0.1824	151.7	w/o view conditioning	0.0632	0.1713			
T2Bs	<b>0.0880</b>	59.6	w/o RefineMLP	0.0613	0.1851	w/o RefineMLP	0.0537	0.2430
T2Bs (SV4D)	0.0882	<b>56.6</b>	[89]	0.0696	0.2654			

Table 1. **(Left)** Quantitative comparison of 4D generation on per-frame image quality when changing the source of multi-view video. T2Bs with the source of both 4Real-Video (T2Bs) and SV4D (T2Bs(SV4D)) achieve high-quality results and improve significantly from the inputs. **(Middle)** Effect of each proposed component on geometry accuracy and smoothness. **(Right)** Effect of RefineMLP on appearance inconsistencies **across time**. Abbreviation: p2p - point to point euclidean distance, NC - normal consistency.

											
LPIPS	plant	bug	shark	alien	indoraptor	deer	cryinghead	corgi	robot	fox	average
SV4D	0.2405	0.2270	0.0928	0.1356	0.1681	0.1168	0.1441	0.1078	0.1887	0.1219	0.1543
4Real-Video	0.2096	0.1986	0.1627	0.1872	0.1754	0.1761	0.1797	0.1624	0.1759	0.1960	0.1824
T2Bs	<b>0.0938</b>	<b>0.1365</b>	0.0610	0.1065	0.0995	<b>0.0697</b>	<b>0.0562</b>	0.0681	0.1065	0.0817	<b>0.0880</b>
T2Bs (SV4D)	0.1388	0.1734	<b>0.0454</b>	<b>0.0989</b>	<b>0.0910</b>	0.0787	0.0575	<b>0.0357</b>	<b>0.0943</b>	<b>0.0679</b>	<b>0.0882</b>

FID	plant	bug	shark	alien	indoraptor	deer	cryinghead	corgi	robot	fox	average
SV4D	179.2400	293.5054	49.7282	144.4213	141.5446	166.7694	171.7080	28.2255	411.4537	86.6902	167.3286
4Real-Video	112.5859	240.1668	69.2304	168.9460	199.0181	214.8099	174.2843	50.8184	191.6526	95.3293	151.6842
T2Bs	<b>32.5861</b>	125.0183	<b>29.5620</b>	<b>69.0403</b>	<b>80.6074</b>	<b>49.4793</b>	<b>34.8517</b>	20.5047	103.4165	51.2718	59.6338
T2Bs (SV4D)	48.3140	<b>94.9072</b>	32.0060	74.2109	101.4071	55.2417	35.6081	<b>14.8072</b>	<b>65.7210</b>	<b>43.6929</b>	<b>56.5916</b>

p2p	plant	bug	shark	alien	indoraptor	deer	cryinghead	corgi	robot	fox	average
T2Bs	<b>0.0529</b>	<b>0.1243</b>	<b>0.0603</b>	<b>0.0398</b>	0.0400	<b>0.1056</b>	<b>0.0174</b>	<b>0.0189</b>	<b>0.0439</b>	<b>0.0731</b>	<b>0.0576</b>
[89]	0.0543	0.1553	0.0755	0.0550	<b>0.0365</b>	0.1173	0.0230	0.0259	0.0577	0.0951	0.0696

NC	plant	bug	shark	alien	indoraptor	deer	cryinghead	corgi	robot	fox	average
T2Bs	<b>0.0874</b>	<b>0.2207</b>	<b>0.2070</b>	<b>0.0745</b>	<b>0.2087</b>	<b>0.0715</b>	<b>0.2465</b>	<b>0.1450</b>	<b>0.1606</b>	<b>0.1895</b>	<b>0.1611</b>
[89]	0.1593	0.3805	0.2952	0.1156	0.3145	0.3562	0.3486	0.1790	0.1917	0.3129	0.2654

Table 2. (Top) Each sample we use from Objaverse-XL dataset. (Bottom) Per-identity improvement in LPIPS, FID and geometry improvement compared to [89]. Abbreviation: p2p - point to point euclidean distance, NC - normal consistency.

#### 4. Ablation studies on Objaverse-XL samples

We further evaluate our method on 10 artist-animated virtual characters that is closely aligned with our application domain from Objaverse-XL. For each identity, we baked geometry sequences with a shared texture. We rendered (ambient=1.0) fixed-time, fixed-view videos as the pipeline input, and full sequences across all times and views as ground truth.

Table 1 (Left) shows 4D generation quality when switching source multi-view videos from 4Real-Video (T2Bs) to SV4D (T2Bs), which corresponds to Fig. 7. Table 1 (Left) shows the effect of view conditioning, RefineMLP and integrating Linear Blend Skinning (LBS), which correspond to Fig. 6, 8, 9.

Specifically, as for RefineMLP, in order to solve the con-

cern that RefineMLP might be exploited to compensate for geometry, but with GT geometry, we show RefineMLP actually improves the geometry in Tab. 1 (Middle). We attribute this to RefineMLP effectively handling **appearance inconsistencies across views** from 4D generation, since we render input fixed-view videos without appearance inconsistency. To further evaluate RefineMLP, we simulate **appearance inconsistencies across time** by multiplying extreme noise  $\mathcal{U}(0.5, 1.5)$  to the texture map, clamped to  $[0, 1]$ . Instead of running 4D generation, we render the geometry sequence with different noisy textures, ensuring there is no inconsistency across views. As shown in Tab. 1 (Right), RefineMLP still improves geometry quality.

The data of animatable 3D animal head model is limited even in Objaverse-XL. We further shows the per-identity comparison in 4D generation and geometry in Table 2. It's

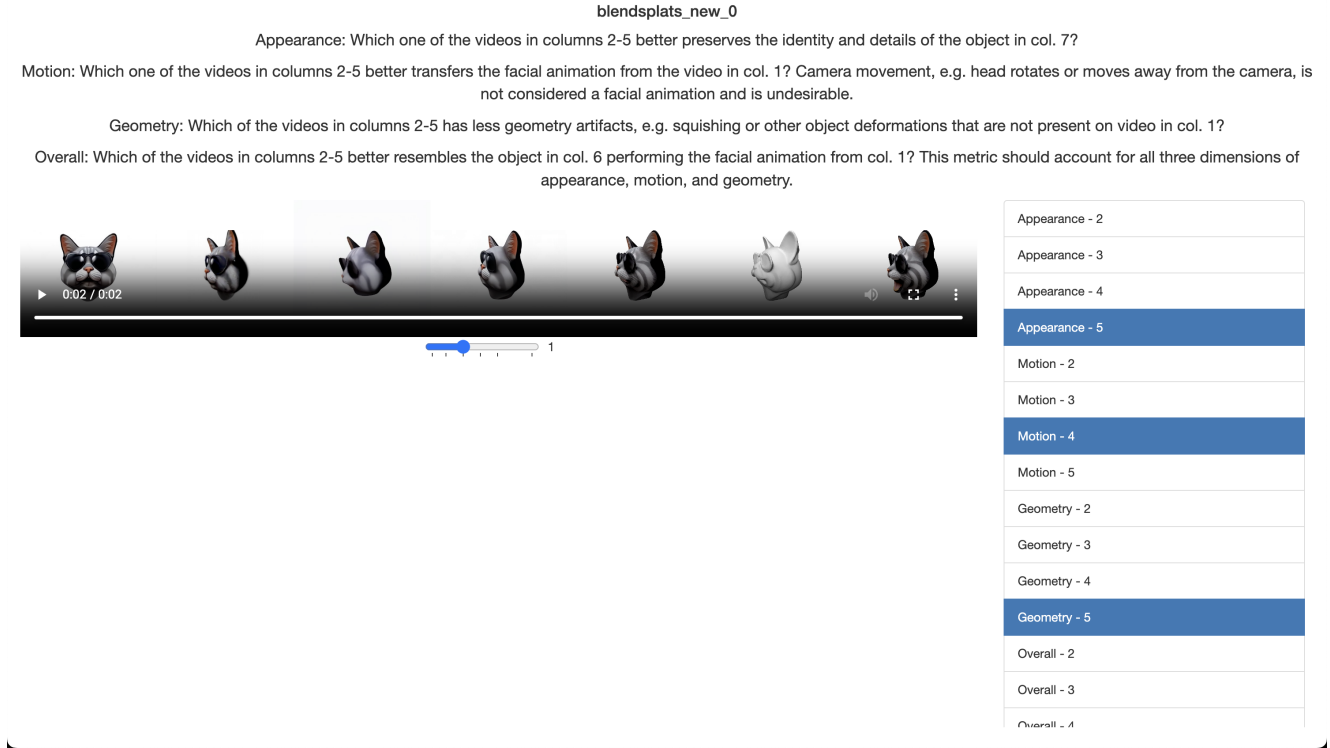


Figure 3. A screenshot of the user study interface. Participants are presented with a set of four single-choice questions, each designed to identify the best-performing column for a given dimension. The data is randomly shuffled to mitigate any potential ordering bias.

clear to see that T2Bs improve significantly on **each** identity.

## 5. User Study Interface

We demonstrate the user interface of our user study in Fig 3. We provide participants with video comparisons of VCDGS and baseline methods. They are free to replay the videos until they make their judgments, selecting the four best-performing columns based on four different criteria. Participants can also use the provided slider to zoom in and out, especially to zoom in for detailed appearance differences.