GeoAware3D: Enhancing Semantic Features for 3D Shape Decoration

Marco Börner   Jed Guzelkabaagac
Technical University of Munich
{marco.boerner, jed.guzelkabaagac}@tum.de

Abstract

We introduce GeoAware3D, a 3D feature descriptor designed for meshes and point clouds. Our approach is class-agnostic, requiring no explicit labelling, and is applicable to diverse 3D shapes. Leveraging robust 2D foundational models, namely Diffusion and DINO, we extract features and project them back into the 3D domain. By incorporating advancements in image-based semantic correspondence, our method enhances geometric awareness leading to precise and descriptive features. Notably, our method requires no training or additional data, and is more simplistic than its predecessors. We demonstrate its efficacy through benchmarks on SHREC'19, highlighting its expressive capabilities.

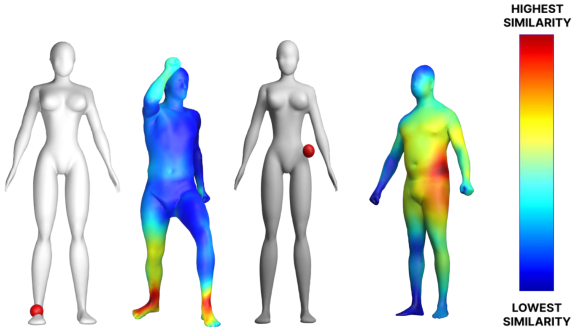

Figure 1. **Semantic Correspondence with GeoAware3D.** Unlike shape correspondence, which categorizes points and faces of meshes into discrete classes, we generate high-dimensional semantic features for each point. This is a fundamentally more challenging task with numerous downstream applications.

1. Introduction

Extracting meaningful features in three dimensions is a fundamental and challenging task in computer vision. Given a pair of 3D meshes, our objective is to decorate the geometric representations with meaningful features. This task is essential for multiple downstream tasks, such as shape correspondence, 3D generative models, and shape morphing.

For decades, exploring geometric invariances [6, 13] was the leading approach for building feature descriptors. Researchers focused on methods that could capture the essential geometric properties of shapes in a way that remained consistent under transformations such as rotation, scaling, and translation.

Additionally, optimization approaches [24] enhanced the robustness and accuracy of these descriptors. Researchers continued to refine the process of matching and recognizing shapes, improving the ability to handle variations in shape and noise. Despite big leaps in the field, generalization remained poor.

Recently, the emergence of deep learning has overhauled past approaches in image analysis tasks. Large-scale foundational models have implicitly learned descriptive and robust semantic features, commonly surpassing previous methods without handcrafting. As outlined in [32], Stable Diffusion features [22] have strong spatial understanding and can generate smooth correspondences, while DINO [17] features provide sparse but accurate matches. Once combined, the feature maps of the two foundational models perform very impressively on downstream tasks.

Our approach utilizes projective analysis [24] by encoding meshes as a set of image projections, performing analysis in 2D, and then unprojecting back into 3D. As previously discussed, this method offers numerous advantages, primarily leveraging powerful pre-trained image models.

Our models lack texture, which is an important challenge to be addressed. Image-based foundational models are generally trained on photorealistic images which are inherently very different and contain more signal. Consequently, we first texture renderings of the mesh using Stable Diffusion, with the 2D rendering serving as guidance for ControlNet [34]. Surprisingly, even with inconsistent texturings from different viewing angles, we still obtain robust features [5] capturing the semantics and geometry.

2. Related Work

This section summarizes fundamentally different approaches to 3D shape correspondence, focusing primarily

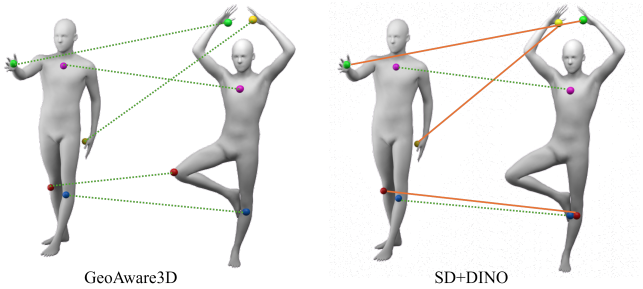

Figure 2. **Qualitative comparison.** Green lines indicate correct matches and red incorrect. Our method can build 'geometrically aware' semantic correspondence even for vastly differing poses, while standard SD+DINO struggles with geometric ambiguity. This distinction is particularly important when decorating human and animal meshes due to the large number of symmetries.

on deep learning-based methods. For a detailed examination of earlier techniques in registration and similarity, refer to Sahillioğlu's comprehensive survey [20].

2.1. Point-to-Point Shape Correspondence

Point-to-point shape correspondence methods aim to map points between surfaces rather than continuous surfaces. For instance, 3D-Coded [7] learns transformations between point clouds by deforming a template shape. Many algorithms like DCP [26], RPMNet [30], and GeomFMap [4] rely on ground truth or mesh connectivity, which can be challenging to obtain.

Recent approaches focus on unsupervised learning on point clouds. CorrNet3D [31] utilizes DGCNN [28] for feature embeddings and includes a symmetry deformation module for reconstruction and correspondence. SE-ORNet [2] aligns point clouds using an orientation module and employs a teacher-student model along with DGCNN for finding correspondences. However, these methods often prioritize geometry and may overlook semantic features.

2.2. Surface Map Methods

Surface map methods simplify shape correspondence between arbitrary 2-manifold surfaces. Traditionally, these methods either leverage eigenfunctions to create functional mappings or construct atlases from $\mathbb{R}^2 \rightarrow \mathbb{R}^n$ (where $n = 2, 3$). They aim to preserve specific geometric properties such as angle preservation for conformal maps. The underlying idea is to map both surfaces onto a common base domain, whether a mesh or a planar region, to facilitate the mapping process.

For example, SURFMNet extends FMNet to handle unsupervised scenarios by enforcing desired structural properties on the resulting functional maps [9, 12]. Unlike traditional methods that require 2-manifold meshes, our approach can establish correspondences even in meshes with imperfections.

2.3. Multi-view Rendering

Multi-view rendering-based learning has proven effective across various 3D tasks. This method, exemplified by projective analysis [27], involves representing shapes through 2D projections, analyzing them in image space, and reintegrating findings into 3D. It has demonstrated strong performance in shape/object recognition [25, 29], human pose estimation [11], reconstruction [18], and segmentation [21].

Specifically, the technique involves rendering 3D shapes from multiple viewpoints and extracting visual descriptors via CNNs trained in supervised settings. Methods for aggregating features across views include averaging, max pooling [23], concatenation, or employing additional CNNs for fusion.

One such method employed by Abdelreheem et al. [1] explores zero-shot correspondence using language models. This approach leverages LLMs and vision models to generate segmentation maps and semantic mappings. Once these maps are obtained, they utilise the 3D semantic segmentation model SAM [10] to get the final geometric descriptors. Due to the reasoning capabilities of language models, this method is particularly effective for strongly non-isometric shape pairs.

Inspired by Dutt et al. [5], who refined projective analysis on meshes using shape renders from diverse angles, our work aims to enhance and simplify their process. Like Dutt et al., we employ zero-shot feature decoration using texturing, diffusion, and DINO to generate geometric and semantic descriptors. However, recent advancements in building features from foundational 2D models offer the potential for more descriptive and consistent 3D features, which we aim to take advantage of.

3. Method

Pre-trained foundational vision models exhibit a behavior where they assign semantic features to pixels in the input image. To accurately de-noise images, the model has to distinguish between adjacent pixels, which is crucial for executing essential downstream tasks such as object detection and image segmentation.

Our central approach leverages this behavior by utilizing a pre-trained foundational model to de-noise renderings, which necessitates the creation of semantic per-pixel features. During this process, the model generates detailed semantic information for each pixel, which we can then extract and transform into a three-dimensional representation.

3.1. Rendering and Texturing

For each mesh we start by representing the 3D model as a collection of k renderings, where viewpoints are sampled

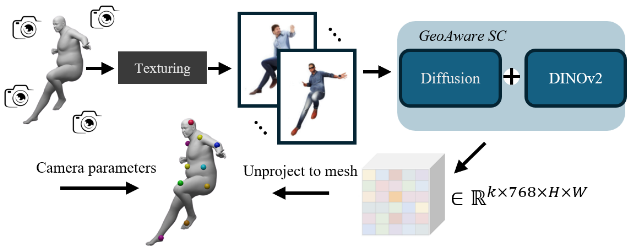

Figure 3. **Pipeline.** GeoAware3D decorates shapes in a zero-shot manner. We take 2D renderings from multiple angles and add texture using ControlNet [34]. We then combine SD and DINO features as outlined in GeoAware SC[33], before aggregating the features back onto the mesh.

uniformly around the shape for a wide coverage. We set the elevation to zero and calculate the azimuths depending on the number of views, taking renderings around the unit sphere.

We then add additional signal to the images by texturing them using ControlNet [34] conditioned on the renderings and prompts such as ”high definition, photo-realistic”. An important thing to note is that meshes may have inconsistently textured images, as shown in 3, however the associated image features are robust and can be aggregated across views [5].

3.2. Diffusion and DINO Features

Visual foundation models such as DINO [17] and SD [19] have demonstrated that the features they learn through self-supervised learning or generative tasks [14, 22] can effectively serve as descriptors for semantic matching, often outperforming previous methods designed specifically for this purpose. Despite this progress these models still have shortcomings [8] in grasping the intrinsic geometry of instances.

3.2.1 Diffusion Hyperfeatures

Various approaches exist to effectively combine SD and DINO features. One such approach by Luo et al. [14] proposes a learned feature aggregator that weights all features, distilling them into a concise descriptor map. Previous methods, which used hand-selected subsets of intermediate diffusion features, are sub-optimal due to the large size and varying semantics of feature maps across different layers and time-steps. Luo et al.'s approach involves a lightweight aggregation network that learns the relative importance of feature maps dependent on the downstream application, aiming to improve tasks like semantic correspondence.

In our case, this method underperformed due to its lack of geometric awareness, which is a crucial factor in human mesh applications.

3.2.2 Geometry-Aware Semantic Correspondence

Zhang et al. [33] enhance the geometric awareness in fused SD+DINO features by fine-tuning an aggregation network that receives the fused features as input and generates a lower-dimensional feature vector for each pixel. They employ pose-variant data augmentation during fine-tuning and use a dense loss additionally to the sparse contrastive loss. Their approach results in a new state-of-the-art for semantic correspondence on the SPair-71k dataset [16], making it a promising candidate for our pipeline.

3.3. Unprojection and View Aggregation

In order to convert the pixel-wise features obtained by foundational images models in 2D space back into 3D, we define an unprojection function U as follows:

$$U := \left(\underset{\text{pixels}}{\mathbb{R}^2} \times \underset{\text{features}}{\mathbb{R}^d}\right) \times \underset{\text{depth}}{\mathbb{R}} \times K \rightarrow \underset{\text{points}}{\mathbb{R}^3} \times \underset{\text{features}}{\mathbb{R}^d}. \quad (1)$$

The function maps pixel-wise features, represented by the pixel locations and features with dimension d, together with the depth of each pixel and the intrinsic camera settings K back into 3D. We apply U on all pixel-wise features extracted from k-views and concatenate all 3D features to obtain a final point cloud P_F with point-wise features.

Since these point-wise features could vary dependent on the view angle they came from, we aggregate them to obtain vertex-wise features:

$$\Phi_{\text{mean}}(v,k) = \frac{1}{k} \sum_{p_f \in N_k(v)} p_f \quad (2)$$

The aggregation function Φ will compute the mean-aggregated features for vertex v using the k nearest point features p_f given by a k-d tree query function N_k. We tried different aggregation methods such as max-pooling or distance weighted mean aggregation but found that simple mean aggregation performs the best.

3.4. Computing Correspondence

There are two ways to compute correspondence. Given two sets of points S and T, sampled from the source mesh and target mesh respectively, we can either use the closest vertex for each point and compute similarities between vertex-wise features or alternatively, apply Φ directly to the sampled points to compute point-wise aggregated features and subsequently the similarities.

We measure similarity between two normalized features f_1 and f_2 using the dot product, which is equal to the cosine similarity.

3.4.1 Closest Vertex

For each source point $s \in S$, we find its closest vertex v_s, forming a set of closest vertices $V_S \subset V$. Let $F_S \in \mathbb{R}^{n\times d}$

be the feature matrix of these n vertices, and $F_T \in \mathbb{R}^{m \times d}$ be the feature matrix of all m vertices in the target mesh.

For each source vertex feature F_{S_i}, the target vertex index j with the highest similarity is given by:

$$j = \text{argmax}_k \; F_{S_i} F_{T_k}^{\top}. \tag{3}$$

We then pick the closest point in the sampled target points set T to the vertex v_j as the predicted corresponding point. A downside of this method is that predictions are restricted to vertices, hence may not be exact point correspondences.

3.4.2 Direct Point-to-Point Correspondence

We directly compute point-wise aggregated features for sampled source and target points S and T, leveraging exact point correspondences. Given the point clouds S and T, we aggregate features using function Φ_{mean} as defined in 2 to obtain feature matrices $F_S \in \mathbb{R}^{n \times d}$ and $F_T \in \mathbb{R}^{m \times d}$. We then compute the similarity matrix $F_S F_T^{\top}$. The highest similarity index j is determined by Eq. 3, allowing precise point-to-point correspondences between the meshes.

4. Evaluation

In order to evaluate our method and to ensure comparability with other methods, we pick the correspondence accuracy metric, which calculates the percentage of predicted points that lie within a certain error tolerance γ. It is given by

$$acc(P) = \frac{1}{n} \sum_{p_s,\, p_t \in P} \mathbb{1}(\|f(p_s) - p_t\|_2 < \gamma d), \tag{4}$$

where $P \in \mathbb{R}^3 \times \mathbb{R}^3$ is the ground truth point-to-point correspondence pairs, $f(\cdot)$ is the location of the predicted corresponding point, γ is the error tolerance and d is the maximum distance between any two points in the target mesh. We follow Dutt et al. [5] and set $\gamma = 0.01$ for a 1% error tolerance.

4.1. Benchmarks

We evaluate and compare our approach on the SHREC'19 dataset [15], which contains a set of 44 human shapes in different poses and 430 ground-truth correspondence annotations. Due to the fact that we opted for the GeoAware approach rather late and have limited compute, we focus on human shapes.

4.2. Results

The results presented in Table 1 highlight the performance of various methods on the SHREC'19 dataset [15], measured in terms of accuracy and runtime per mesh. DIFF3F [5] achieves the highest accuracy at 26.41% but has a moderate runtime of 4.42 minutes per mesh. SE-ORNet (trained) [3] achieves an accuracy of 21.41%, although its runtime data is unavailable in the current table. Our proposed method, GeoAware3D, attains an accuracy of 23.42%, which, while not surpassing the state-of-the-art accuracy of DIFF3F, demonstrates a significant improvement in runtime efficiency, requiring only 1.02 minutes per mesh. GeoAware3D provides a trade-off between accuracy and computational efficiency, outperforming SE-ORNet in accuracy and achieving a faster runtime compared to DIFF3F.

Table 1. **Results on the SHREC'19 dataset.** Accuracy (acc ↑) and runtime (min ↓) in minutes per mesh are reported.

Method	**Accuracy (acc ↑)**	**Runtime (min ↓)**
DIFF3F	**26.41**	4.42
SE-ORNet	21.41	?
GeoAware3D	23.42	**1.02**

Runtime is measured under same system conditions on an A100 GPU using the code published by Luo et al. [5] for DIFF3F.

Table 2. **Ablation study on SHREC'19 dataset.** Accuracy (acc ↑) metric is reported.

Ablation	**SHREC'19 (acc ↑)**
Correspondence in 2D	16.12
Standard SD+DINO	17.81
Hyperfeatures	18.54
GeoAware3D (k-views=32)	23.35
GeoAware3D (k-views=64)	**23.42**

Ablation study evaluating different components and configurations of the GeoAware3D method on the SHREC'19 dataset.

4.3. Ablations

We conduct an ablation study on the SHREC'19 dataset to evaluate the impact of different components in our GeoAware3D method, as shown in Table 2.

The "Correspondence in 2D" strategy bypasses feature aggregation, treating the task as point-to-point correspondence in 2D. This approach necessitates rendering k different views of the mesh, but it selects only the most suitable view for each point targeted for correspondence matching. "Standard SD+DINO" uses stable diffusion and DINO features without any aggregation network. "Hyperfeatures" employs the aggregation network proposed by Luo et al. [14]. Further, we find that our method works well with a low number of k-views, whereas DIFF3F uses 100 views. Doubling the number of views used improved relative accuracy by merely 0.3% .

References

[1] Ahmed Abdelreheem, Abdelrahman Eldesokey, Maks Ovsjanikov, and Peter Wonka. Zero-shot 3d shape correspondence, 2023. 2

[2] Shuxiao Chen, Shunbo Lu, Xiang Bai, and Serge Belongie. Se-ornet: Self-ensemble orientation estimation for point cloud alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6146–6155, 2021. 2

[3] Jiacheng Deng, Chuxin Wang, Jiahao Lu, Jianfeng He, Tianzhu Zhang, Jiyang Yu, and Zhe Zhang. Se-ornet: Self-ensembling orientation-aware network for unsupervised point cloud shape correspondence, 2023. 4

[4] Jacopo Donati and Maks Ovsjanikov. Geomfmap: Robust feature matching using geometric properties of 3d shapes. In *Computer Graphics Forum*, pages 213–225. Wiley Online Library, 2021. 2

[5] Niladri Shekhar Dutt, Sanjeev Muralikrishnan, and Niloy J. Mitra. Diffusion 3d features (diff3f): Decorating untextured shapes with distilled semantic features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4494–4504, 2024. 1, 2, 3, 4

[6] Marvin Eisenberger, Zorah Lähner, and Daniel Cremers. Divergence-free shape interpolation and correspondence. *ArXiv*, abs/1806.10417, 2018. 1

[7] Thibault Groueix, Matthew Fisher, Vladimir G Kim, Bryan C Russell, and Mathieu Aubry. 3d-coded: 3d correspondences by deep deformation. In *European Conference on Computer Vision*, pages 230–246. Springer, 2018. 2

[8] Kamal Gupta, Varun Jampani, Carlos Esteves, Abhinav Shrivastava, Ameesh Makadia, Noah Snavely, and Abhishek Kar. Asic: Aligning sparse in-the-wild image collections, 2023. 3

[9] Ohad Halimi, Emanuele Rodola, Or Litany, Alex Bronstein, and Ron Kimmel. Surfmnet: Learning surfel-based features for robust functional maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(5):1727–1738, 2021. 2

[10] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023. 2

[11] Jiahao Lin and Gim Hee Lee. Multi-view multi-person 3d pose estimation with plane sweep stereo, 2021. 2

[12] Or Litany, Alexander M Bronstein, Michael M Bronstein, and Xiang Bai. Fmnet: Learning feature metric networks for 3d shape analysis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 587–595, 2017. 2

[13] Or Litany, Tal Remez, Emanuele Rodolà, Alexander M. Bronstein, and Michael M. Bronstein. Deep functional maps: Structured prediction for dense shape correspondence. *CoRR*, abs/1704.08686, 2017. 1

[14] Grace Luo, Lisa Dunlap, Dong Huk Park, Aleksander Holynski, and Trevor Darrell. Diffusion hyperfeatures: Searching through time and space for semantic correspondence, 2024. 3, 4

[15] S. Melzi, R. Marin, E. Rodolà, U. Castellani, J. Ren, A. Poulenard, P. Wonka, and M. Ovsjanikov. Shrec'19: Matching humans with different connectivity. In *Proc. EUROGRAPHICS Workshop on 3D Object Retrieval (3DOR)*, Genova, Italy, 2019. [Contest Report]. 4

[16] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Spair-71k: A large-scale benchmark for semantic correspondence. *arXiv preprint arXiv:1908.10543*, 2019. 3

[17] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. 1, 3

[18] Yvain Quéau, Jean Mélou, Jean-Denis Durou, and Daniel Cremers. Dense multi-view 3d-reconstruction without dense correspondences, 2017. 2

[19] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. 3

[20] Yusuf Sahillioğlu. Recent advances in shape correspondence. *The Visual Computer*, 36, 2020. 2

[21] Gopal Sharma, Kangxue Yin, Subhransu Maji, Evangelos Kalogerakis, Or Litany, and Sanja Fidler. Mvdecor: Multi-view dense correspondence learning for fine-grained 3d segmentation, 2022. 2

[22] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion, 2023. 1, 3

[23] Bach Tran, Binh-Son Hua, Anh Tuan Tran, and Minh Hoai. Self-supervised learning with multi-view rendering for 3d point cloud analysis, 2022. 2

[24] Oliver van Kaick, Ghassan Hamarneh, and Daniel Cohen-Or. A survey on shape correspondence. *Comput. Graph. Forum*, 30:1681–1707, 2011. 1

[25] Wenju Wang, Yu Cai, and Tao Wang. Multi-view dual attention network for 3d object recognition. *Neural Computing and Applications*, 34:1–12, 2022. 2

[26] Yue Wang and Justin M Solomon. Dcp: Deep closest point. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3523–3532, 2019. 2

[27] Yunhai Wang, Minglun Gong, Tianhua Wang, Daniel Cohen-Or, and Baoquan Chen. Projective analysis for 3d shape segmentation. *ACM Transactions on Graphics*, 32:1–12, 2013. 2

[28] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. In *ACM Transactions on Graphics (TOG)*, pages 1–12. ACM New York, NY, USA, 2019. 2

[29] Yong Xu, Chaoda Zheng, Ruotao Xu, Yuhui Quan, and Haibin Ling. Multi-view 3d shape recognition via correspondence-aware deep learning. *IEEE Transactions on Image Processing*, 30:5299–5312, 2021. 2

[30] Zi Jian Yew and Gim Hee Lee. Rpm-net: Robust point matching using learned features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11824–11833, 2020. 2

[31] Wei Zeng, Hang Yang, Siyu Zhu, Jiaya Liu, Bo Dai, and Yu Qiao. Cornet3d: Unsupervised end-to-end learning of dense correspondence for 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6052–6061, 2021. 2

[32] Junyi Zhang, Charles Herrmann, Junhwa Hur, Luisa Polania Cabrera, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence. 2023. 1

[33] Junyi Zhang, Charles Herrmann, Junhwa Hur, Eric Chen, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. Telling left from right: Identifying geometry-aware semantic correspondence, 2024. 3

[34] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 1, 3