Electron Density-enhanced Molecular Geometry Learning

Hongxin Xiang[1] , **Jun Xia**[2] , **Xin Jin**[3] , **Wenjie Du**[4] , **Li Zeng**[1] and **Xiangxiang Zeng**[1,*]

[1]College of Computer Science and Electronic Engineering, Hunan University, Changsha, China
[2]School of Engineering, Westlake University, Hangzhou, China
[3]Eastern Institute of Technology, Ningbo, China
[4]University of Science and Technology of China, Hefei, China

[*]Correspondding author (xzeng@hnu.edu.cn)

A Effect of grid spacing on electron density calculation

The electron density distribution is continuous in space, but in numerical calculations, it must be discretized and represented as data points on a three-dimensional grid. Therefore, during the electron density calculation stage based on density functional theory (DFT), the grid spacing parameter defines the spacing between grid points. A smaller grid spacing can better capture the changes in electron density. However, the computation time and memory requirements rise sharply with the increase in the number of grid points.

Here, we show the results of DFT-based electron density calculations for the Pepcid[1] molecule (35 atoms in total) using different grid spacings. As shown in Talbe S1, as expected, as the grid spacing becomes smaller, the number of points in the electron density, storage space occupation, and calculation time become significantly larger. In particular, when the grid spacing is equal to 0.05, the number of points is as high as tens of millions, the storage space is close to 1G, and the calculation time is close to 4 hours. This inspires us to set the grid spacing reasonably to achieve a balance between accuracy and efficiency.

Table S1: Number of data points, disk storage space, and time spent performing DFT-based electron density calculations for the Pepcid molecule with different grid spacings. #Points refers to the number of points in the electron density.

Grid Spacing	#Points	Disk Storage (KB)	Time Cost (s)
0.05	73,243,170	941,768	13,130.53
0.1	9,239,840	118,809	1,987.14
0.2	1,183,680	15,222	588.88
0.3	362,664	4,666	444.01
0.4	155,595	2,003	402.86
0.5	81,312	1,048	397.12
0.6	48,692	628	390.95
0.7	30,720	397	379.44
0.8	20,580	267	383.96
0.9	15,200	198	383.94
1	11,339	148	381.23

B Experiment comparison of point cloud, voxel, and the proposed ED images

To verify the effectiveness of different ED representations, we randomly selected 10,000 molecular ED information and corresponding 6 energy-related labels from 2 million DFT data as our dataset. For convenience, we refer to it as DB_{ED}-1w. In DB_{ED}-1w, there are 6 important quantum chemical properties related to energy, namely:

- **DF-RKS Final Energy** describes the stability of molecular systems and is used to calculate thermodynamic properties such as heat of reaction and free energy change.
- **Nuclear Repulsion Energy** represents the Coulomb interaction between nuclei and reflects the contribution of the geometric structure of the nucleus to the molecular energy.
- **One-Electron Energy** includes the interaction between electrons and nuclei and the kinetic energy of electrons, which is crucial in the analysis of molecular orbitals and electron distribution.
- **Two-Electron Energy** describe the interaction between electrons, reflecting important information related to electrons. It is of significant significance for the analysis of the properties of the electronic structure and the correlation energy of the system.
- **DFT Exchange-Correlation Energy** is an approximate representation of electron exchange and correlation interactions in density functional theory (DFT) and has a direct impact on the total energy and ED distribution.
- **Total Energy** represents the total energy of a system, which is central to the study of thermodynamics and kinetics, especially when comparing molecular stabilities or studying reaction pathways.

Subsequently, we represent the ED information in the DFT data as point cloud $\mathcal{PC} \in \mathbb{R}^{n^v \times 4}$ (n^v represents the number of points in the ED and 4 represents the coordinates x, y, z and the corresponding ED intensity), voxel $\mathcal{VO} \in \mathbb{R}^{60\times60\times60}$ and images $\mathcal{U} \in \mathbb{R}^{6\times4\times224\times224}$ respectively. Therefore, in order to fairly compare the most basic capabilities of different representations, we select the classic model in the corresponding field for each representation, which are point cloud-based PointNet [Qi *et al.*, 2017], voxel-based ResNet3D [Hara *et al.*, 2018] and image-based ResNet18 [He *et al.*, 2016] respectively.

[1] https://pubchem.ncbi.nlm.nih.gov/compound/3325

C The details of more background

Electron density (ED) is a key bridge to understand the prediction of energy and forces in molecular force fields (MFF), which uses $p(r)$ to describe the distribution of electrons in space, where r represents the position vector in space, usually expressed in Cartesian coordinates (x, y, z) or spherical coordinates (r, θ, ϕ). The core task of MFF is to predict the total energy and interatomic forces of the system by describing the interactions between atoms, thereby simulating molecular behavior and dynamic processes. ED provides complete information about the distribution of electrons in molecules, which can directly reflect the bonding interactions, non-bonded interactions (such as van der Waals forces and hydrogen bonds) between atoms, and charge distribution characteristics. Through ED, different components of energy (such as electron kinetic energy, exchange-correlation energy, etc.) can be accurately calculated from a physical level. These energy components are the basis of molecular force field models and are used to describe the potential for interactions between atoms. Therefore, ED is not only the core output of quantum mechanical calculations, but also provides a solid theoretical basis for constructing high-precision MFF and understanding complex interactions between molecules. This further demonstrates the significance of the proposed method to introduce ED into the learning framework of geometry representation.

Next, we introduce the B3LYP and basis set of 6-31G**/+G** used in this paper. B3LYP and 6-31G**(+G**) play an important role in ED descriptions. They directly affect the accuracy and efficiency of the calculation results. B3LYP is a more widely used hybrid functional in which the exchange energy is combined with the exact energy from Hartree-Fock theory. It is widely used in chemical reactions, molecular orbital analysis, and calculations of intermolecular forces. 6-31G**/+G** is a commonly used basis set used to represent the mathematical expression of atomic orbitals.

D The overall process of EDG

In order to clearly describe the proposed EDG framework, we show the overall process in Algorithm 1, which includes 3 stages.

Algorithm 1 The overall process of ED-enhanced molecular geometry representation learning framework (EDG).

Data: The molecular geometry $\mathcal{G}$, corresponding ground-truth labels y; the multi-views ED images $\mathcal{U}$ and the multi-view structural images $\mathcal{S}$.
Stage I: ED Representation Learning with ImageED: Each view image from ED images $\mathcal{U}$ is divided into multiple patches and each patch is regarded as a token. Subsequently, we mask 25% of the tokens and input the unmasked tokens into the ED encoder f_{EDE} to extract the latent features of the tokens. Then, we use additional features initialized by 0 to serve as the features of the unmasked tokens and concatenate it with the latent features of the unmasked tokens. Finally, we input them into the ED decoder f_{EDD} to predict all masked tokens and restore all unmasked tokens.
Stage II: Pre-training of ED-aware Teacher: Structural images $\mathcal{S}$ and ED images $\mathcal{U}$ are input into the ED-aware teacher f_S and the frozen ED encoder f_{EDE} to extract structural features $\mathcal{F}^{\mathcal{S}} = f_S(\mathcal{S})$ and ED features $\mathcal{F}^{\mathcal{U}} = f_{EDE}(\mathcal{U})$, respectively. Subsequently, $\mathcal{F}^{\mathcal{S}}$ is further input into an ED predictor f_{EDP} to map the structural features into ED features $\mathcal{F}^{\mathcal{S}\to\mathcal{U}} = f_{EDP}(\mathcal{F}^{\mathcal{S}})$. Finally, f_S and f_{EDP} will be supervised by aligning $\mathcal{F}^{\mathcal{S}\to\mathcal{U}}$ and $\mathcal{F}^{\mathcal{U}}$.
Stage III: ED-enhanced Molecular Geometry Learning: During training stage, we feed the molecular geometry $\mathcal{G}$ and structural images $\mathcal{S}$ into the trainable geometry student f_G and the frozen ED-aware teacher to extract geometry features $\mathcal{F}^{\mathcal{G}}$ and structural features $\mathcal{F}^{\mathcal{S}}$, respectively. Subsequently, $\mathcal{F}^{\mathcal{G}}$ is further fed into the mapper f_M to obtain structural features $\mathcal{F}^{\mathcal{G}\to\mathcal{S}}$. $\mathcal{F}^{\mathcal{G}}$ is fed into the task predictor and supervised by the task-related labels y. $\mathcal{F}^{\mathcal{S}}$ and $\mathcal{F}^{\mathcal{G}\to\mathcal{S}}$ are input into the frozen ED predictor to extract ED features $\mathcal{F}^{\mathcal{S}\to\mathcal{U}}$ and $\mathcal{F}^{\mathcal{G}\to\mathcal{U}}$ and guided by $\mathcal{F}^{\mathcal{S}\to\mathcal{U}}$. During inference stage, we only input molecular geometry into geometry student and task predictor to get the corresponding prediction results.

E Rendering details of ED image

We use PyMol [DeLano and others, 2002] to render ED images. In EDG, we propose ED loader, structural loader and multi-view joint render to generate ED images. We describe their pseudocode in detail as follows:

- **ED loader.** We show the PyMol pseudocode of ED loader in Algorithm 2. In detail, in #3, $[-0.08, 0, 0.08]$ and [red, white, blue] represent the threshold used to map colors and the range of color filtering, respectively, where -0.08 0, 0.08 represent the most negative potential value (red), the neutral potential value (white), and the most positive potential value, respectively. The purpose of setting 0.08 is to make the color mapping smoother. In #4, the isosurface represents to set isovalue of surface to 0.05, where 0.05 is a common threshold value indicating that we use an ED value of 0.05 to generate an equipotential surface. The isovalue is the threshold of ED value, meaning that only those regions with ED greater than 0.05 will be visualized, thus highlighting the high-density regions in the molecular structure. Finally, in #5, we use ESP to color map the surface in the ED. The reason for using ESP is that it more directly reflects the electrical characteristics of the molecular surface area and has better physical and chemical interpretation.
- **Structural loader.** We use PyMol command *load* {*conformation file*} to load a conformation image from a conformation file.

- **Multi-View Joint Render.** We show the PyMol pseudocode of multi-view joint render in Algorithm 3. By modifying the defined $axis$ and $angle$, we can generate multi-view ED images.

Algorithm 2 The PyMol command of ED loader for loading ED information.

Input: ED files $files_{ED}$, ESP files $files_{ESP}$
for sampled a ED file $file_{ED}$ from $files_{ED}$ and a ESP file $file_{ESP}$ from $files_{ESP}$ **do**
  #1 Load ED file from path and name it ED
  load $file_{ED}$, ED
  #2 Load ESP file from path name it ESP
  load $file_{ESP}$, ESP
  #3 Set a color map, called legend, that maps values ([-0.08, 0, 0.08]) of ESP to a color gradient ([red, white, blue]) to visually represent different ESP values in an image.
  ramp_new legend, ESP, [-0.08, 0, 0.08], [red, white, blue]
  #4 Create an isosurface named surface using the ED data with an isovalue of 0.05.
  isosurface surface, ED, 0.05
  #5 Use the surface_color command to set the color of the ED surface as the legend.
  set surface_color, legend, surface
end for

Algorithm 3 The main PyMol command of multi-view joint render for rendering ED images.

Input: The save path $path$, height $height$ and width $width$ of ED images, the axis $axis$ and angle $angle$ of view rotation
for sampled a ED file $file_{conf}$ from $files_{conf}$ **do**
  #1 Set the transparency of the ED image to 0.4
  set transparency, 0.4
  #2 Rotate the image by $angle$ degrees along the $axis$ axis
  turn $axis$, $angle$
  #3 Renders the image and saves it to $path$ with a width of $width$ and a height of $height$
  png $path$, width=$width$, height=$height$
end for

F Selection of ED-aware teachers

To learn the mapping from readily available molecular conformations to ED information, we experimentally validate the ability of two conformation-based modalities in predicting ED, namely conformational geometry, which uses atomic coordinates and atom-to-atom edges to represent conformation, and structural images, which directly represent conformation as images. Specifically, we use structural loader and structural render to generate 4-views of structural images with $(x, 0), (x, 180), (y, 180), (z, 180)$ as the rotation axis and rotation angle of the view.

For efficiency, we sample 10,000 conformations from 2 million pre-training dataset for experiments. We use the ED encoder from frozen ImageED to extract ED features $\mathcal{F}^{\mathcal{U}}$ from multi-view ED images and these features are viewed as the ground-truth of ED information. We randomly split the training/validation/test sets in a ratio of 8/1/1. Subsequently, we select SchNet [Schütt *et al.*, 2017], SE3-Transformer [Fuchs *et al.*, 2020], EGNN [Satorras *et al.*, 2021] to predict ED features $\mathcal{F}^{\mathcal{G}\to\mathcal{U}}$ from conformational geometry and use ResNet18 [He *et al.*, 2016] to predict ED features $\mathcal{F}^{\mathcal{S}\to\mathcal{U}}$ from multi-view structural images. Finally, we use the MAE metric to calculate the gap between $(\mathcal{F}^{\mathcal{U}}, \mathcal{F}^{\mathcal{G}\to\mathcal{U}})$ and $(\mathcal{F}^{\mathcal{U}}, \mathcal{F}^{\mathcal{S}\to\mathcal{U}})$. If the gap is smaller, the ability to extract ED information is stronger.

To be fair, we use the same experimental settings to train all models. Specifically, we use a batch size of 8 and a learning rate of 0.001 for 30 epochs. As shown in Table S2, we find that ResNet18 based on structural image achieves the best performance with an relative improvement of 11.5% compared with the methods based on conformational geometry, which indicates that using structural image to predict ED features is a better choice. Therefore, we use ResNet18 based on structural image as ED-aware teacher.

G The details pf pre-training ImageED

We describe the ImageED pre-training loss in Figure S1, which is pre-trained with 20 epochs (about 21k steps). As shown in the figure, the losses $\mathcal{L}_{MP}$ and $\mathcal{L}_{RP}$ of ImageED gradually decrease and converge as the training progresses, which indicates that ImageED is well trained and can learn ED-related knowledge from ED images.

Table S2: MAE performance of different methods in predicting ED features, which is calculated by using the ED features predicted by different methods and the ground-truth ED features from ED encoder of ImageED with MAE metric. Δ represents the relative performance improvement of ResNet18 based on structural image compared with other best results. **Bold** and underlined indicate the best and second best results respectively.

Conformational Modalities	**Model**	**MAE**
Conformational Geometry	SchNet	<u>0.02737</u>
	SE3-Transformer	0.02853
	EGNN	0.04096
Structural Image	ResNet18	**0.02421**
-	Δ	$\uparrow$11.5%

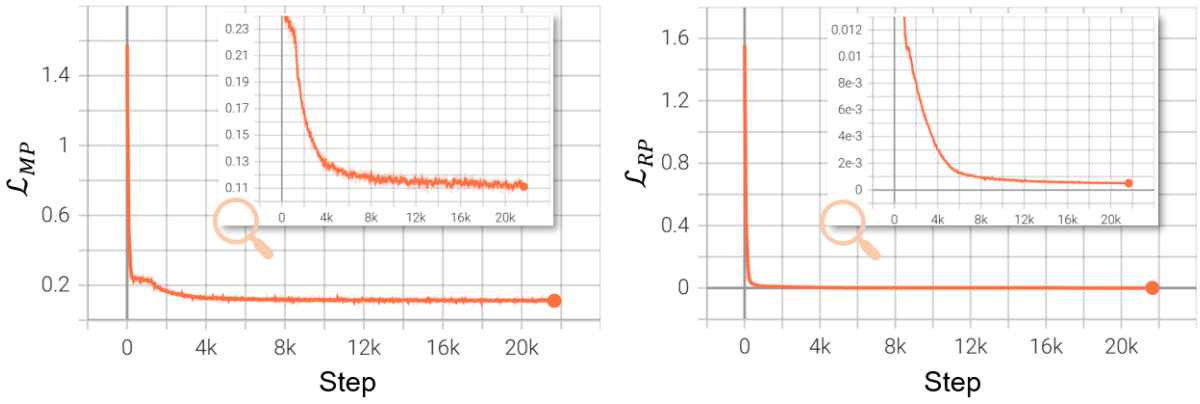

Figure S1: The pre-training losses $\mathcal{L}_{MP}$ and $\mathcal{L}_{RP}$ of ImageED. The x-axis and y-axis represent the pre-training step and the corresponding loss, respectively. The magnifying glass shows the loss figure at a larger scale.

Furthermore, in order to observe the differences in the ED images of different molecules, we randomly sample 10,000 molecular pairs (M_i, M_j) from our pre-training dataset. Note that the molecules in each molecular pair consist of 6-view RGB-D electron density images. Subsequently, we use the encoder f_{EDE} in ImageED to extract ED features $f_{EDE}(M_i)$ and $f_{EDE}(M_j)$ from these molecular pairs. Finally, we calculate the cosine similarity and Euclidean distance between these paired features, respectively. As shown in Figure S2, we find that the distribution of Euc distance is much flatter than that of Cosine similarity, which indicates that the differences can be eliminated in Euc distance. Therefore, this motivates us to use the Euc distance-based loss to train the ED-aware teacher.

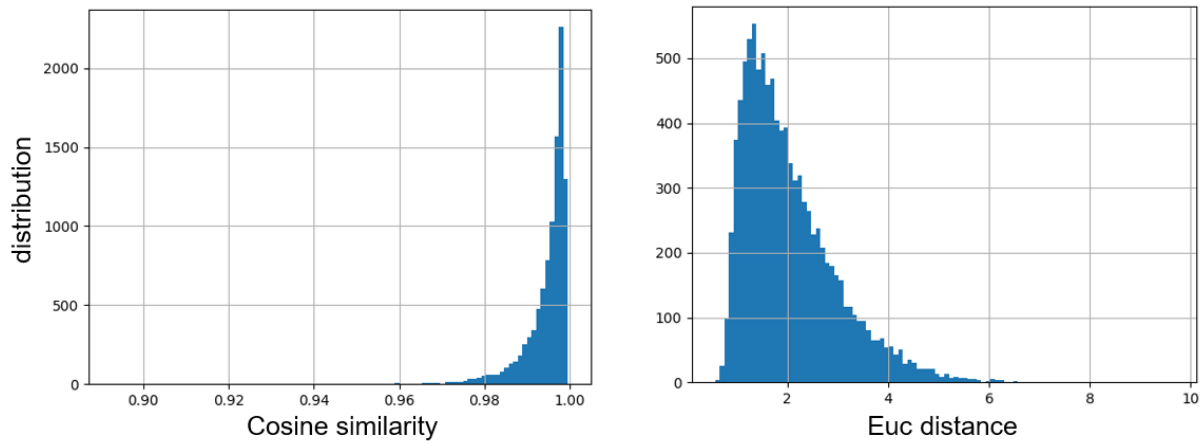

Figure S2: Euclidean distance and cosine similarity distribution of 10,000 pairs of ED images using ImageED. Euc distance represents Euclidean distance.

H The details of pre-training ED-aware teacher

In the pre-training of ED-aware teacher, we use 2% of the data as validation set and the remaining 98% of the data as training set. Figure S3 shows the loss of ED-aware teacher on the training set and validation set during pre-training. We find that the ED-aware teacher has a low loss on training set and validation set, which indicates that ED-related knowledge from ImageED can be well learned by the ED-aware teacher.

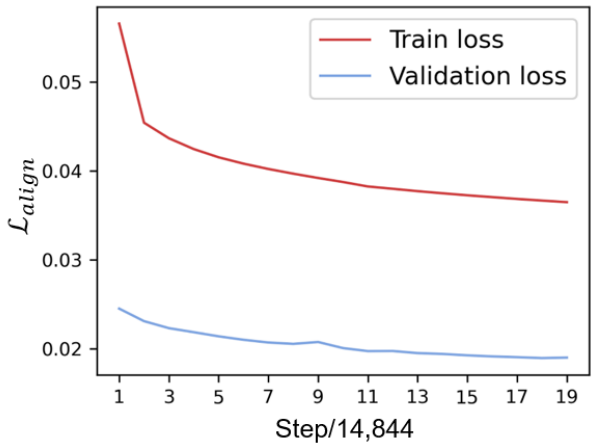

Figure S3: The pre-training loss $\mathcal{L}_{align}$ of ED-ware teacher. The x-axis and y-axis represent the pre-training step and the corresponding loss, respectively. It is worth noting that the x-axis step size is divided by 14,844 for the sake of presentation.

I More examples about visualization of ImageED

We show more visualization examples of image restoration using ImageED and ImageED w/o $\mathcal{L}_{RP}$. Figure S4 and Figure S5 show the RGB output and RGB-D output from ED encoder of ImageED. We summarize two main findings: (1) ImageED is able

to recover the overall contour and texture of ED images from latent features, which indicates that ImageED can learn ED-related knowledge; (2) ImageED w/o $\mathcal{L}_{EP}$ is limited in restoring ED images, especially in RGB-D output, ImageED w/o $\mathcal{L}_{EP}$ does not contain any texture-related pixels, which indicates that $\mathcal{L}_{EP}$ is necessary for ImageED to learn rich ED-related information.

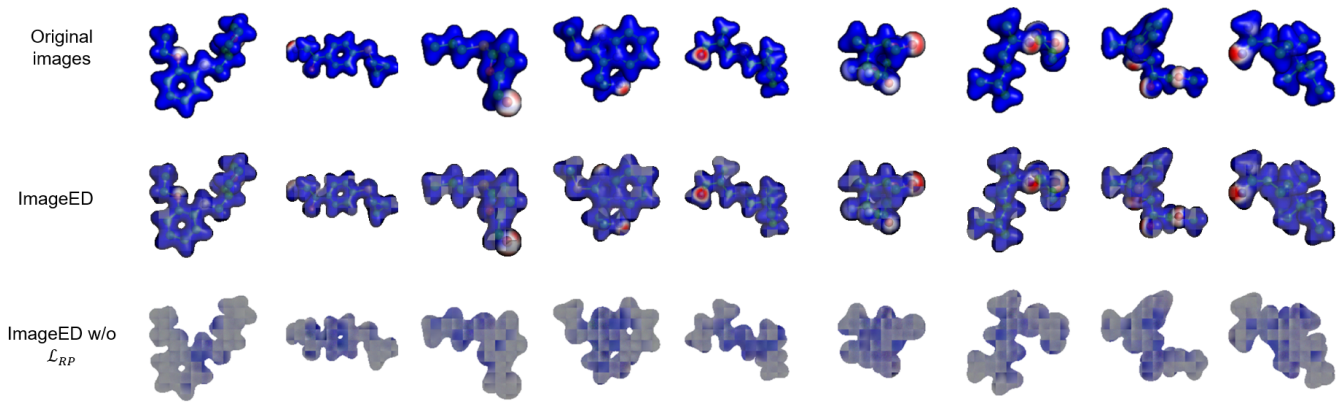

Figure S4: Visualization of the output of the ED encoder from ImageED. We only show the R, G, B channels.

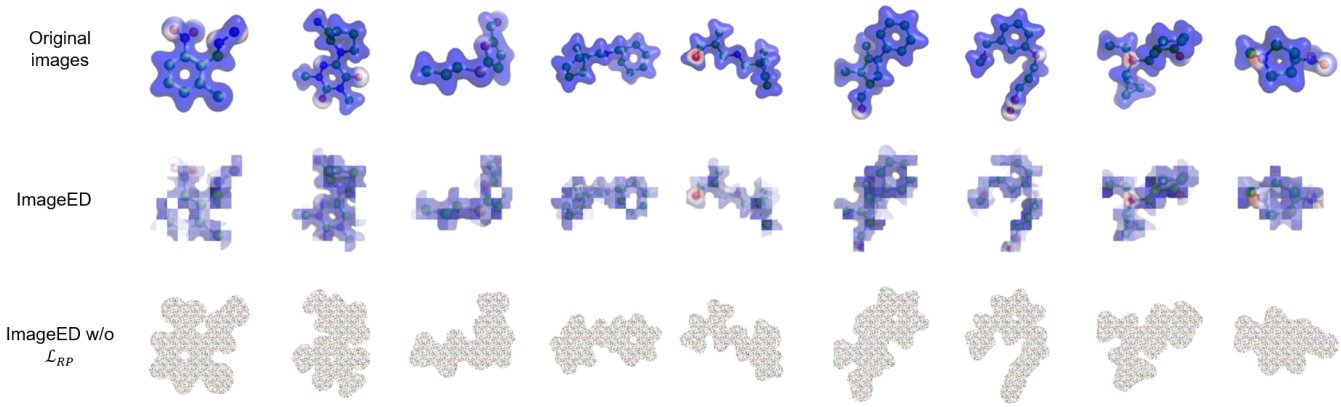

Figure S5: Visualization of the output of the ED encoder from ImageED. We show all R, G, B, D channels.

References

[DeLano and others, 2002] Warren L DeLano et al. Pymol: An open-source molecular graphics tool. *CCP4 Newsl. Protein Crystallogr*, 40(1):82–92, 2002.

[Fuchs *et al.*, 2020] Fabian Fuchs, Daniel Worrall, Volker Fischer, and Max Welling. Se (3)-transformers: 3d roto-translation equivariant attention networks. *Advances in neural information processing systems*, 33:1970–1981, 2020.

[Hara *et al.*, 2018] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6546–6555, 2018.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[Qi *et al.*, 2017] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[Satorras *et al.*, 2021] Vıctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. In *International conference on machine learning*, pages 9323–9332. PMLR, 2021.

[Schütt *et al.*, 2017] Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *Advances in neural information processing systems*, 30, 2017.