First of all, we would like to thank the reviewers for their constructive feedback and thorough reviews. We will include detailed remarks and the references proposed by **R3** in the final version of the paper. Our method combines a spectral convolution feature extractor with a hierarchical, fully differentiable matching layer based on entropy regularized optimal transport and an unsupervised loss. As the reviewers acknowledged, our formulation leads to an increased accuracy and decreased computational cost, even in comparison with sota supervised methods.

Novelty (R2, R3) Our formulation follows the line of work pioneered by FMNet [4] where the idea is to build an axiomatic matching method as a layer into a NN. Although there has been an abundance of follow-up work [6,7,8], our approach is the first to use a more elaborate matching layer than the standard functional maps (FM) proposed in [4]. Our OT matching layer uses both extrinsic and intrinsic embedding information and processes the input features in a coarse-to-fine manner. Moreover, we are the first to combine a spectral CNN feature extractor with an axiomatic matching method. In comparison to prior work, our network is more accurate and generalizes better across benchmarks.

Comparison with GeoFMNet (R3) We believe that there was a fundamental misunderstanding with regards to our results in Table 1 and we would like to clarify this, as we believe that the main concerns of **R3** under 3.2, 3.3, 3.5, 4., 5. and 8. boil down to this: **R3** states that "dataset[s] with different triangulation where SHOT based methods fail badly [...] are not tested at all in this paper" and that "[in order] to test [the generalization to unseen datasets with different triangulation], GeoFMNet proposed three benchmark settings with different triangulations which are ignored in this paper". In this context, *we want to strongly emphasize that all the experiments in Table 1 are performed on the remeshed versions of FAUST and SCAPE*. These datasets indeed contain shapes with varying triangulation, therefore our results in Table 1 prove that our method is robust to this type of input noise. Moreover, we not only show results on the individual two datasets but also four more settings where we test the generalization between the different datasets. For these results, the meshing is again different and the SHOT descriptors are even less reliable due to varying local features, see Table 1. The remaining "3 of 5" experiments in [7, Fig. 3.] that **R3** frequently refers to are based on Surreal which is essentially a superset of FAUST with the same SMPL triangulation, similar local features and much more poses. For us, the additional value is marginal, e.g. the performance of GeoFMNet on Sur.-F/Sur.-S [7, Figure 3] and F-F/F-S [7, Table 1] are almost identical. We will state this point more clearly in the paper and thank the reviewer for the insight. For our experimental setup on FAUST and SCAPE remeshed we followed the standard protocol in this line of work [4,6,7,8].

SHOT descriptors ↔ PC feature extractor (R1,R3) To date, there are two orthogonal approaches to extract learned local features on 3D shapes: Treating the input shapes as an unordered collection of SHOT feature vectors [4,6,8] or as 3D point clouds [7,30]. We agree with the reviewers that SHOT descriptors are suboptimal since they are local, unstable and triangulation-dependent. However, for our purposes, we still prefer the former approach for multiple reasons: Most sota PC feature extractors are not invariant to rotations or near-isometries which is highly unnatural for 3D surfaces. According to [7, Appendix C] this leads to problems for humans in "bent over poses", see also Figure 2 of our Appendix. We believe the drastic improvements observed in Table 1 of our Appendix confirm the value of the proposed spectral convolution layer in aggregating information in the neighborhood of each point and thereby boosting the precision and robustness of the method such that even with a noisy local descriptor we achieve state-of-the-art performance.

Detailed remarks (R2) 1) Fig. 3. shows a comparison of the relative conformal distortion of triangles [40, Eq. (3)]. Indeed, our method is similar to [9] in terms of this error metric. In contrast to [9], we make use of a spectral convolution layer that drastically reduces the number of large-scale mismatches, see Fig. 4 and Table 1. The curves do not saturate because we evaluate the distortion on the remeshed datasets – even for a perfect matching some triangles get distorted. We will include the ground-truth curves in the plot for reference and thank the reviewer for the remark.
2) These two datasets are indeed very relevant for shape correspondence, however, it is FAUST re and SCAPE re which are to date widely accepted as the standard benchmarks for learning based shape matching methods for the following reasons: SHREC'19 contains only 44 shapes with severely varying poses and non-isometries to a degree that prohibits a meaningful train/test split (in GeoFMNet, the authors use the easier, remeshed version of SHREC'19 where all shapes have approx. the same resolution). For **R3**: The same holds true for SHREC'16 which only contains 25 shapes. The FAUST online challenge is at this point saturated with high-performing methods that specialize on humans, e.g. the public results from 3D coded [30] and smooth shells [9] involve using a human template which gives them an unfair advantage over true general-purpose matching methods. We suspect, that this is the reason why the most recent supervised and unsupervised methods [7] and [6] also refrained from evaluating themselves on this online challenge.
3) Our method implicitly assumes shapes with bounded distortion. This means that, like smooth shells [9], our method will fail for extremely non-isometric pairs (topological changes, partiality, ...), but this is even more true for the learning methods that are based on functional maps [4,6,7,8] which strongly favor nearly-isometric pairs. Regarding the "resistance [...] to 3D misalignment", our approach is invariant to rigid poses of the shapes and we set the scaling of the inputs to a fixed square root area of $\frac{2}{3}$.
4) We use the code from the authors' github pages [28,29] with a default number of eigenfunctions of 120.