



# ShapeKit

Junqi Liu, Dongli He, Wenxuan Li, Ningyu Wang, Alan L. Yuille,  
and Zongwei Zhou<sup>(✉)</sup>

Johns Hopkins University, Baltimore, USA  
zzhou82@jh.edu

**Abstract.** In this paper, we present a practical approach to improve anatomical shape accuracy in whole-body medical segmentation. Our analysis shows that a shape-focused toolkit can enhance segmentation performance by over 8%—without the need for model re-training or fine-tuning. In comparison, modifications to model architecture typically lead to marginal gains of less than 3%. Motivated by this observation, we introduce ShapeKit, a flexible and easy-to-integrate toolkit designed to refine anatomical shapes. We evaluate our method on two large-scale, multi-institutional CT scan datasets with expert-verified annotations. This work highlights the underappreciated value of shape-based tools and calls attention to their potential impact within the medical segmentation community.

ShapeKit is available at <https://github.com/BodyMaps/ShapeKit>.

**Keywords:** Shapes · Anatomical Structures · Quality Control · Toolkit

## 1 Introduction

Medical image segmentation sits at the heart of many clinical and research workflows because it turns raw images into *measurable* anatomy and pathology [23, 38]. Modern deep networks can already outline dozens of organs in a single pass, yet their *raw* masks still carry avoidable defects—tiny speckles, broken boundaries, swapped left-and-right labels, or anatomically impossible shapes [2, 18, 20, 29]. These irregularities lower segmentation accuracy, distort volumetric measurements, and may trigger erroneous downstream decisions [21, 26, 32].

Most research tackles these errors by redesigning network architectures or fine-tuning models [15, 24, 25, 37]. Unfortunately, such changes often deliver only marginal gains (typically <3% DSC scores if all learning parameters are sufficiently optimized) [1, 16]. In contrast, a much simpler lever anatomical-aware post-processing remains under-explored. We show that correcting masks *after* inference can lift multi-organ DSC score by >8% without touching the network weights. This striking gap motivates a fresh look at post-processing as a first-class citizen in the segmentation pipeline.

J. Liu and D. He—Equal contribution. Authors are permitted to list their names first in their CVs.

We introduce **ShapeKit**, an easy-to-plug-in toolkit that fixes common shape mistakes in whole-body CT masks. **ShapeKit** offers:

- **A taxonomy of shape errors** (artifacts, false positives, fragmented structures, redundant lobes, laterality swaps) distilled from large-scale studies and reader feedback, guiding systematic correction rather than ad-hoc rules.
- **General and organ-specific fixes** built from fast morphological and topological operators that harmonize labels, reconnect broken parts, delete spurious voxels, and re-assert anatomical plausibility.
- **Scalability out of the box**: parallel batch processing, vectorized NumPy/SciPy kernels, and minimal memory overhead enable thousands of 3-D volumes to be cleaned in minutes on a workstation.

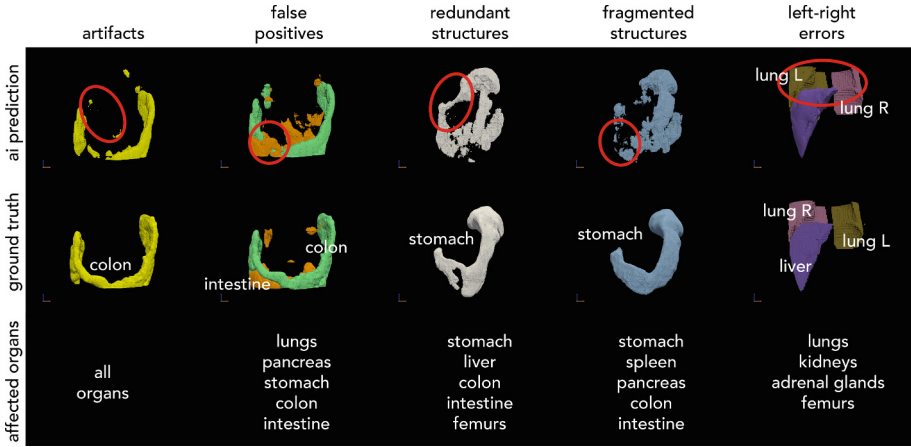
Performing solely on the predicted masks, **ShapeKit** is *model-agnostic*. It can upgrade any public or proprietary segmentation models, old or new, with zero re-training cost—an attractive property for clinical deployment where model updates are tightly regulated. More broadly, our results rekindle interest in shape reasoning, echoing earlier work on rule-based and morphological clean-ups [9, 30] while extending them into a unified, reproducible framework that scales to heterogeneous multi-organ datasets.

1. We quantify how much pure anatomical-aware post-processing can surpass architecture tweaks, delivering >10% DSC improvement in widely-adopted, highly-competitive segmentation benchmark.
2. We release **ShapeKit**, an open-source, highly configurable toolkit that standardizes and refines whole-body CT masks at scale.
3. We provide an extensive error taxonomy and ablation study to guide future work on shape-aware medical segmentation.

## 2 Related Work

Post-processing is a critical but often under-emphasized step in medical image segmentation. Early techniques relied heavily on manual correction and hand-crafted rules to eliminate artifacts and fix mislabeled regions [31]. With the rise of deep learning, most effort shifted toward improving segmentation networks [36, 39], yet the final predictions from even the strongest models frequently suffer from shape errors—such as disconnected fragments, implausible geometry, or label confusion—that require correction downstream [26].

Recent toolkits, such as nnU-Net [15], MONAI [4], and DAP-Atlas [8], provide basic post-processing utilities, including connected component filtering and label remapping. However, these modules are often fixed in design and lack anatomical reasoning. More advanced efforts have explored shape priors [13] or statistical harmonization [9, 33], but these are usually domain-specific or computationally intensive. Despite growing datasets and deployment in clinical workflows, few systems offer a general, extensible, and anatomically aware post-processing pipeline capable of handling diverse whole-body CT data.



**Fig. 1.** Types and distribution of common segmentation errors identified across different abdominal organs in the VISTA3D dataset. Categories include false positives, redundant or fragmented structures, and left-right errors. Organs are grouped by frequency of impact.

ShapeKit addresses this gap. Unlike prior toolkits, it introduces a detailed taxonomy of shape errors, a set of modular correction functions, and both general-purpose and organ-specific refinements—all with scalability and ease-of-use in mind. This toolkit is designed not only to clean segmentation outputs but also to enforce anatomical plausibility at scale.

### 3 ShapeKit

ShapeKit is a lightweight, modular toolkit for anatomical shape correction in CT segmentation. It offers two tiers of functions: general-purpose operations for common cleanup tasks and organ-specific refinements for anatomical accuracy. The entire system is plug-and-play, requiring no retraining or model changes.

#### 3.1 A Taxonomy of Shape Errors

Segmentation outputs often contain structural inconsistencies that deviate from expected anatomical patterns. These shape errors are not just cosmetic—they can distort volumetric measurements, hinder morphometric analysis, and mislead downstream algorithms. Based on analysis of large-scale whole-body CT datasets, including expert feedback and manual inspection, we categorize the most frequent error modes into five types, visualized in Fig. 1. To ensure the credibility of our findings, we involved ten board-certified radiologists from Johns Hopkins University. The inspection process combined direct evaluation of CT scans with 3D visualization tools, enabling thorough assessment of anatomical consistency and segmentation quality.

1. **Artifacts:** These appear as small, isolated voxel clusters that float outside the anatomical region of interest. For example, speckle-like debris near the liver or stomach (Fig. 1, column 1) often arise due to imaging noise or minor model misfires. While small, these anomalies can falsely inflate organ volumes and confuse spatial reasoning tasks.
2. **False Positives:** These are anatomically implausible regions labeled as an organ, but which actually belong to unrelated tissue. As shown in Fig. 1 (column 2), a portion of the intestines is mistakenly labeled as colon, or parts of the stomach leak into surrounding space. These misclassifications are often more prominent in low-contrast areas or near organ boundaries.
3. **Redundant Structures:** These errors extend the predicted shape into abnormal or biologically impossible regions. For instance, the segmentation may include duplicated lobes of the lung, or extensions of the colon that protrude into the abdominal cavity without anatomical basis (Fig. 1, column 3). Such predictions violate known organ morphology and must be removed or corrected.
4. **Fragmented Structures:** Rather than being detected as a single continuous object, the organ is split into disconnected fragments. Figure 1 (column 4) shows this clearly in the pancreas and colon—organs that are anatomically continuous but predicted as multiple unconnected blobs. This affects both the anatomical integrity and the reliability of region-based statistics.
5. **Left-Right Errors:** Perhaps the most dangerous type of mistake in clinical contexts, these occur when left/right anatomical labels are swapped or missing entirely. In Fig. 1 (column 5), the left and right lungs are either merged or mislabeled, and the liver appears on the wrong side. Such spatial inversion could severely impact tasks like surgical planning or disease localization.

By systematically identifying these categories, ShapeKit enables a structured and reproducible approach to post-processing, moving beyond ad-hoc filtering or patch-based fixes. Each of the functions is designed to directly address one or more of these common shape inconsistencies.

### 3.2 General ShapeKit Functions

ShapeKit provides a suite of general-purpose functions designed to correct common structural errors in organ segmentation masks. These functions are model-agnostic and operate directly on binary or multi-label masks produced by any segmentation algorithm. By focusing on structural integrity and anatomical plausibility, they help convert raw predictions into clinically usable outputs. Each function is modular and can be flexibly composed with others, enabling users to tailor the pipeline to different datasets or application needs.

1. **Removal of Small Components.** One of the most frequent segmentation artifacts involves the presence of small, disconnected regions that do not belong to the actual organ. These tiny speckles or blobs are often caused by noise in the image or minor classification errors by the model.

**ShapeKit** removes such components based on a user-defined volume threshold. This ensures that only sufficiently large, anatomically meaningful regions are retained, thereby improving the accuracy and cleanliness of the final mask.

2. **Suppression of Non-Dominant Components.** In cases where multiple disconnected regions of varying sizes are labeled as the same organ, **ShapeKit** retains only the most dominant ones. This function ranks connected components by size and preserves only the largest (e.g., top 1 or top 2), assuming that smaller components are likely to be spurious. This is particularly useful for organs such as the liver or pancreas, which should appear as single contiguous structures.
3. **False Positive Reassignment.** False positives occur when a region is mistakenly labeled as an organ but actually corresponds to nearby tissue. To address this, **ShapeKit** analyzes spatial relationships between predicted regions and known anatomical adjacency. If a component appears too far from the expected location of a given organ or is closer to another organ, it is relabeled accordingly. This function improves specificity and reduces label leakage between adjacent organs, which is especially important in complex regions like the abdomen.
4. **Left–Right Separation and Validation.** For symmetric organs—such as lungs, kidneys, and adrenal glands—laterality matters. **ShapeKit** introduces a spatially aware function to divide merged predictions into left and right components. It first splits the mask along a reference anatomical axis (typically the sagittal plane), then assigns each component to the appropriate side based on spatial proximity to a reliable landmark, such as the liver (assumed to reside on the right side of the body). This prevents left-right confusion and ensures consistency in downstream analysis or reporting.
5. **Merging Fragmented Structures.** Sometimes, an organ is predicted as multiple broken parts that should, in reality, be a single entity. **ShapeKit** detects when multiple disconnected components of the same organ are in close spatial proximity and merges them into a unified structure. This enhances continuity and avoids misleading interpretations of organ shape, size, or completeness.

Together, these general functions address the majority of shape errors that plague raw segmentation masks. They require no training or annotation effort and can be applied to any organ or dataset.

### 3.3 Specialized ShapeKit Functions

While general post-processing functions help fix common shape issues across all organs, certain errors are unique to specific anatomical structures. For example, the lungs require careful left–right separation, the intestines are prone to fragmentation, and the pancreas may appear as several disconnected parts due to its irregular shape. To address these challenges, **ShapeKit** offers specialized functions tailored to the geometry and spatial context of individual organs.

These specialized functions build upon the general tools but include additional logic based on organ-specific knowledge. For example, the toolkit recognizes that the liver is always located on the right side of the body. This anatomical fact is used to help resolve left-right confusion in neighboring organs, such as the lungs or kidneys. A few key examples of these specialized functions include:

1. **Left–Right Assignment Using the Liver.** ShapeKit first divides symmetric organs (like the lungs or kidneys) into left and right sides by looking at their location in the scan. But because some patients are positioned differently, this step may not always be accurate. To make it more reliable, ShapeKit compares the position of these organs to the liver. Since the liver is known to sit on the right side of the body, it can act as a reference point to confirm or correct the organ labels. This is especially helpful when both sides of an organ are merged or mislabeled.
2. **Retaining Key Structures While Filtering Noise.** For each organ, ShapeKit includes size and shape thresholds to decide what to keep and what to discard. For instance, when cleaning up the lungs or kidneys, the toolkit keeps only the two largest components (left and right) and removes smaller, likely incorrect pieces. These rules are informed by the typical size and count of organs and help avoid over-cleaning or under-filtering.

Importantly, all these specialized corrections are modular. Users can choose which organs to process and which rules to apply using a configuration file. This flexible setup makes it easy to apply ShapeKit to new datasets or specific clinical tasks without re-writing the code. The modular design also allows users to build custom pipelines by combining only the needed functions, similar to assembling blocks in a Lego set.

In summary, the organ-specific functions bring domain knowledge into the correction process. By understanding how each organ should appear and where it should be located, ShapeKit provides precise and trustworthy refinements that go beyond generic image processing.

### 3.4 Computational Optimization

To ensure scalability across large datasets, ShapeKit incorporates several key computational optimizations.

First, the pipeline supports parallel execution using Python’s multiprocessing, enabling batch processing of thousands of CT volumes. Each scan is handled independently, allowing efficient workload distribution across CPU cores—critical for time-sensitive clinical settings or large-scale studies.

Second, ShapeKit relies on vectorized operations via high-performance libraries such as NumPy and SciPy. These implementations eliminate slow Python loops and leverage optimized C-backed routines for morphological operations, connected component analysis, and spatial filtering.

Third, to reduce memory usage, ShapeKit processes only the relevant masks in compressed data types (e.g., bool, uint8) and clears intermediate arrays after

**Table 1.** Comparison of DSC scores before and after applying ShapeKit for multiple organs. Organs are ordered by the magnitude of improvement ( $\Delta$ ) in descending order.

<b>Dataset A</b>	gall	bladder	pancreas	lung L	stomach	lung R	duodenum
VISTA3D [12]	71.7	79.3	87.5	91.4	89.0	72.9	
VISTA3D+ShapeKit	79.7	84.5	91.0	93.7	91.2	75.0	
$\Delta$	<b>+8.0</b>	<b>+5.2</b>	<b>+3.5</b>	<b>+2.3</b>	<b>+2.2</b>	<b>+2.1</b>	
<b>Dataset B</b>	gall	bladder	pancreas	colon	lung R	lung L	stomach
VISTA3D [12]	68.1	73.5	80.1	88.6	87.8	91.3	
VISTA3D+ShapeKit	76.9	79.0	84.2	91.7	90.5	93.6	
$\Delta$	<b>+8.8</b>	<b>+5.5</b>	<b>+4.1</b>	<b>+3.1</b>	<b>+2.7</b>	<b>+2.3</b>	

use. The system also employs lazy evaluation: if no correction is needed for a particular function, the step is skipped automatically.

Finally, the modular design allows users to enable only selected functions via a configuration file, avoiding unnecessary computation and making the pipeline efficient and adaptable to diverse datasets.

## 4 Experiment and Result

**Dataset.** We used a subset of the JHH dataset [28, 34] (denoted as Dataset A) and a subset of AbdomenAtlasPro [3, 19, 20] (denoted as Dataset B) to evaluate the ShapeKit pipeline. The JHH dataset includes 5,160 contrast-enhanced CT scans (venous and arterial phases) from international sites, centralized at Johns Hopkins Hospital. Each scan contains metadata such as age, race, gender, and diagnosis. Voxel-wise annotations were manually created by radiologists and verified by senior experts. All scans were de-identified and approved for use by the Johns Hopkins IRB (IRB00403268). AbdomenAtlasPro comprises 20,460 abdominal CT volumes from 112 hospitals across diverse populations. It includes 673K segmentation masks across 22 anatomical structures, generated through a two-stage pipeline combining expert annotation and AI-assisted refinement. And all data were anonymized prior to public release. For the purpose of evaluation, representative subsets from both the JHH and AbdomenAtlasPro datasets were selected and exclusively reserved for testing.

**Evaluation Metric.** To systematically evaluate the effectiveness of the ShapeKit pipeline, we employed the Dice–Sørensen coefficient (DSC) [6] as the primary performance metric. For each subject, DSC values were computed organ-wise, both before and after applying the correction functions. This comparative analysis enabled a detailed quantification of segmentation improvements attributable to shape corrections, capturing both global and localized refinements. By assessing changes in DSC across organs, we were able to identify which anatomical structures benefited most from specific correction strategies and to characterize the overall robustness of the pipeline.

## 4.1 Quantitative Results

Table 1 summarizes the Dice–Sørensen Coefficient (DSC) scores across multiple organs before and after applying ShapeKit to segmentation outputs from the VISTA3D model [12]. The evaluation was conducted on two independent test sets, i.e., Dataset A and Dataset B, each sampled from different clinical sources and imaging protocols. Importantly, both datasets are out-of-distribution (OOD) relative to the training data used by VISTA3D, making this a strong test of generalization.

Across both datasets, ShapeKit consistently improves segmentation accuracy, reinforcing its effectiveness as a post-processing layer under distribution shift. In Dataset A, the most notable gains are observed in the gall bladder (+8.0 DSC) and pancreas (+5.2 DSC), organs known to pose challenges due to their small size and variable appearance. Improvements in lung L/R, stomach, and duodenum (ranging from +2.1 to +3.5) suggest that ShapeKit also successfully corrects subtle boundary errors and enforces anatomical symmetry.

In Dataset B, similar trends are observed, with even higher gains in gall bladder (+8.8) and pancreas (+5.5), as well as in more complex, elongated structures like the colon (+4.1). These results indicate that ShapeKit is not only effective in refining predictions from in-distribution data, but also robust to changes in data domain, scanner protocol, or patient population.

One particularly noteworthy observation is that even high-performing organs—such as the lungs and stomach, which already have DSCs exceeding 85%, experience meaningful gains. This suggests that ShapeKit is not simply compensating for weak segmentations, but systematically improving mask plausibility across the board by reducing noise, correcting structural errors, and restoring anatomical integrity.

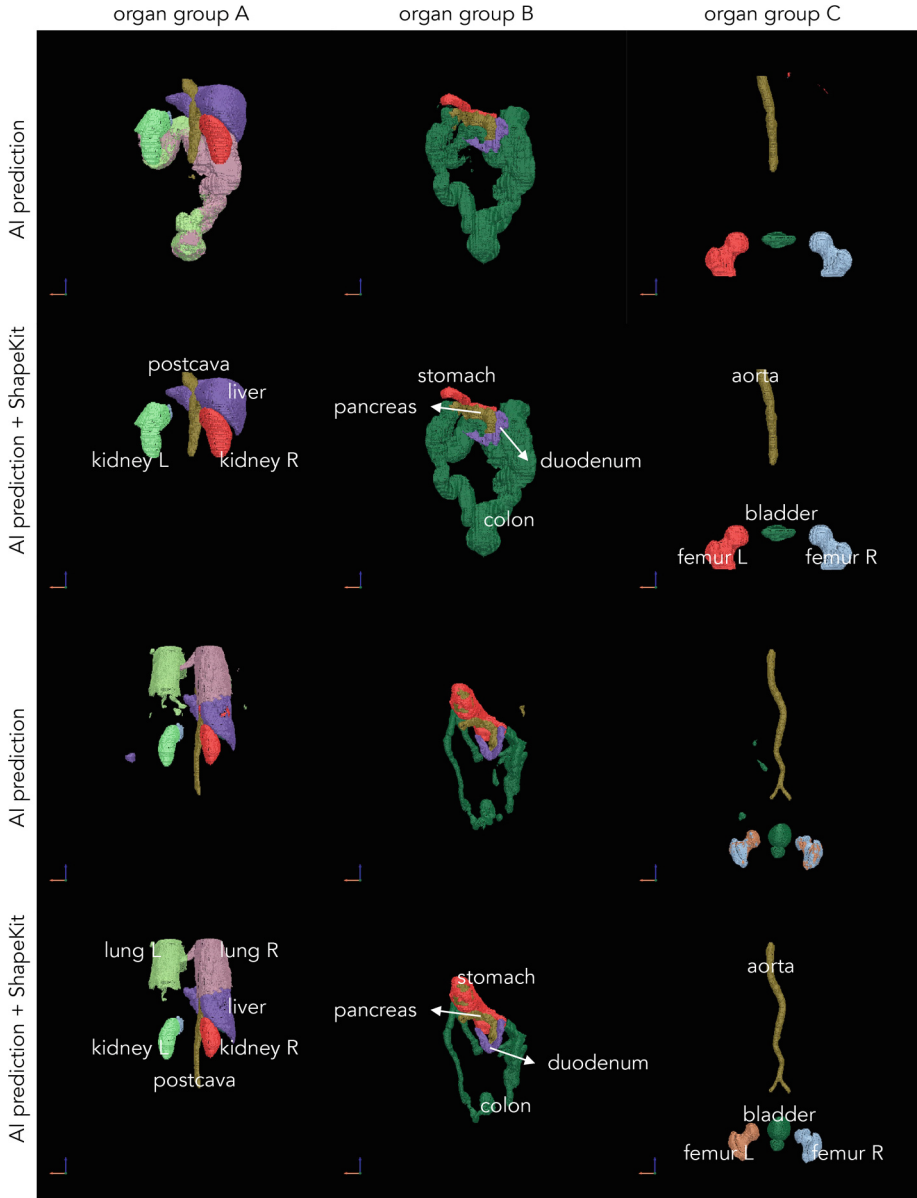
## 4.2 Visualization Results

Figure 2 illustrates qualitative comparisons of segmentation results before and after applying ShapeKit. These examples highlight common failure modes—such as artifacts, false positives, fragmented regions, and left-right label errors—across a variety of organs including the lungs, pancreas, colon, and stomach.

In the raw predictions, we observe noisy speckles around the liver and intestines, discontinuous pancreas masks, and incorrect left-right labeling in symmetric organs like the lungs and kidneys. Such errors degrade anatomical coherence and can mislead downstream measurements, especially for shape-sensitive tasks like volume estimation or surgical planning.

After post-processing with ShapeKit, these issues are significantly mitigated. Fragmented organs are reconnected, redundant or implausible structures are removed, and left/right sides are correctly resolved based on liver positioning. Notably, organs such as the pancreas and colon—which are particularly prone to false positives and fragmentation—show visibly more compact and anatomically realistic segmentations. Improvements in high-performing organs like the lungs





**Fig. 2.** Visualization of two representative cases segmented by ai prediction, with and without ShapeKit. To facilitate comparison, annotated organs are grouped into three categories: Group A: Upper Body Organs (lungs, esophagus, liver, gall bladder, hepatic vessel, portal and splenic veins, kidneys, adrenal glands, postcava), Group B: Digestive Organs (stomach, pancreas, duodenum, colon, intestine, rectum), and Group C: Lower Body Structures (femurs, aorta, celiac trunk, bladder, prostate). In the post-processed segmentations, organ labels are overlaid for clarity.

and stomach further demonstrate that **ShapeKit** refines even strong predictions by enforcing symmetry and boundary regularity.

These visual results confirm that **ShapeKit** not only removes superficial noise but meaningfully improves anatomical plausibility, contributing to more reliable and interpretable outputs for clinical and quantitative analysis.

## 5 Discussion and Conclusion

We introduced **ShapeKit**, a flexible, model-agnostic toolkit that systematically corrects shape-related errors in whole-body CT segmentation. Through a suite of modular correction functions, **ShapeKit** improves anatomical plausibility without requiring model retraining or architectural changes. Our results show consistent DSC improvements in two out-of-distribution datasets, highlighting its robustness and ease of integration into existing segmentation models.

A key insight from this work is that post-processing, often overlooked, can deliver meaningful improvements—especially in challenging or fragmented predictions. **ShapeKit** addresses not only cosmetic issues like speckles or label leakage but also enforces deeper structural consistency, such as left-right correctness and organ continuity.

Importantly, anatomically accurate shape masks are increasingly essential beyond evaluation. Recent efforts in synthetic data generation, such as NVIDIA’s MAISI platform [10, 11], and other related studies [5, 7, 14, 17, 22, 27, 35], rely on precise segmentation shapes to synthesize realistic CT volumes. In these pipelines, toolkits like **ShapeKit** serve as a critical preconditioning step—ensuring that imperfect model outputs do not propagate into downstream training data. Similarly, recent work by Li et al. [18] shows that shape consistency enhances model robustness and domain transfer. **ShapeKit** contributes directly to this by refining outputs in a reproducible, interpretable manner.

Despite its strengths, **ShapeKit** is currently rule-based and may oversimplify segmentations in highly complex structures, such as the colon. Future extensions could integrate learned shape priors or uncertainty-aware corrections to further improve generalization across anatomical variations and pathologies.

In conclusion, **ShapeKit** elevates the role of post-processing in medical image segmentation. By enforcing anatomical structure at scale, it not only enhances segmentation quality but also supports broader applications such as data curation, synthetic image generation, and shape-aware modeling. We release **ShapeKit** as an open-source toolkit and hope it serves as a foundation for future shape-centric advances in medical AI.

**Acknowledgments.** This work was supported by the Lustgarten Foundation for Pancreatic Cancer Research and the National Institutes of Health (NIH) under Award Number R01EB037669. We would like to thank the Johns Hopkins Research IT team in [IT@JH](#) for their support and infrastructure resources where some of these analyses were conducted; especially [DISCOVERY HPC](#).

**Disclosure of Interests.** The authors declare no competing interests.

## Appendix: Major ShapeKit Functions

### Remove Small Components

This function removes small disconnected regions from a binary mask based on a specified volume threshold. It is commonly used to eliminate noisy artifacts, improving the anatomical plausibility of the segmentation.

```
def remove_small_components(
    mask: np.array,
    threshold: int):
    """
    Remove small components based on a given volume threshold.

    Args:
        mask (np.array): Binary 3D array.
        threshold (int): Minimum number of voxels a component
            must have to be kept.

    Returns:
        np.array: A cleaned binary mask.
    """
```

### False Positive Reassignment

This function aims to correct false positives that arise when components are mistakenly assigned to the wrong organ. If a component is found to be closer to a neighboring organ, it is reassigned accordingly. This improves segmentation specificity and reduces cross-organ mislabeling.

```
def reassign_false_positives(
    segmentation_dict: dict,
    organ_adjacency_map: dict,
    check_size_threshold=500):
    """
    Reassign false positives between anatomically adjacent organs.

    Args:
        segmentation_dict (dict): A dictionary mapping organ
            names to binary masks.
        organ_adjacency_map (dict): A dictionary defining
            spatial adjacency between organs.
        check_size_threshold (int, optional): Minimum component
            size to consider for reassignment.

    Returns:
```

```

dict: Updated segmentation dictionary.
"""

```

## Suppress Non-dominant Components

This function retains only the top N largest connected components in a binary mask, removing smaller fragments. It is commonly used to eliminate noise or fragmented structures while preserving the dominant anatomical region.

```

def suppress_non_largest_components_binary(
    mask: np.array,
    keep_top=2):
    """
    Suppress all but the top-N largest connected components
    in a binary mask.

    Args:
        mask (np.ndarray): Binary mask
        keep_top (int): Number of largest components to keep.

    Returns:
        np.ndarray: Cleaned binary mask.
    """

```

## Left-right Division Check

This step separates symmetric organs (e.g. lungs, kidneys) into left and right components based on their spatial position. Components are split along a chosen axis (typically the sagittal plane), and laterality is verified using the anatomical location of a reference organ such as the liver, which is assumed to reside on the right side.

```

def split_right_left(mask, AXIS=0):
    """
    Splits a symmetric organ mask into right and left components
    along a specified axis.

    Each connected component in the mask is assigned to either
    the left or right side based on the mean coordinate of its
    voxels along the specified axis.

    Args:
        mask (np.ndarray): Binary 3D mask containing
        the merged organ(s).
        AXIS (int): Axis along which to perform the split.
        Default is 0 (left-right).

```

Returns:

```
    right_mask (np.ndarray): Binary mask.
    left_mask (np.ndarray): Binary mask.
    """
```

```
def reassign_left_right_based_on_liver(
```

```
    right_mask: np.array,
    left_mask: np.array,
    liver_mask: np.array):
    """
```

Reassign left and right masks based on proximity to the liver.  
Liver is assumed to always be on the right side anatomically.

Args:

```
    right_mask (np.ndarray): Binary mask for the presumed
    right-side organ.
    left_mask (np.ndarray): Binary mask for the presumed
    left-side organ.
    liver_mask (np.ndarray): Binary mask for the liver
    (used as spatial reference).
```

Returns:

```
    corrected_right_mask, corrected_left_mask : np.ndarray
    """
```

## References

1. Bassi, P.R., et al.: Touchstone benchmark: are we on the right way for evaluating ai algorithms for medical segmentation? Conference on Neural Information Processing Systems (2024). <https://github.com/MrGiovanni/Touchstone>
2. Bassi, P.R., et al.: Label critic: Design data before models. arXiv preprint [arXiv:2411.02753](https://arxiv.org/abs/2411.02753) (2024). <https://github.com/PedroRASB/LabelCritic>
3. Bassi, P.R., et al.: Radgpt: constructing 3d image-text tumor datasets. arXiv preprint [arXiv:2501.04678](https://arxiv.org/abs/2501.04678) (2025). <https://github.com/MrGiovanni/RadGPT>
4. Cardoso, M.J et al.: Monai: An open-source framework for deep learning in health-care. arXiv preprint [arXiv:2211.02701](https://arxiv.org/abs/2211.02701) (2022)
5. Chen, Q., et al.: Towards generalizable tumor synthesis. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (2024). <https://github.com/MrGiovanni/DiffTumor>
6. Dice, L.R.: Measures of the amount of ecologic association between species. *Ecology* **26**(3), 297–302 (1945)
7. Du, S., et al.: Boosting dermatoscopic lesion segmentation via diffusion models with visual and textual prompts. In: 2024 IEEE International Symposium on Biomedical Imaging (ISBI), pp. 1–5. IEEE (2024). <https://github.com/zoedisy/BoostingLesionSegviaDiffusionModels>

8. Gatidis, S., et al.: A whole-body fdg-pet/ct dataset with manually annotated tumor lesions. *Scientific Data* **9**(1), 601 (2022)
9. Gibson, E., et al.: Automatic multi-organ segmentation on abdominal ct with dense v-networks. *IEEE Trans. Med. Imaging* **37**(8), 1822–1834 (2018)
10. Guo, P., et al.: Text2ct: Towards 3d ct volume generation from free-text descriptions using diffusion model. *arXiv preprint [arXiv:2505.04522](https://arxiv.org/abs/2505.04522)* (2025)
11. Guo, P., et al.: Maisi: medical ai for synthetic imaging. In: 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 4430–4441. IEEE (2025)
12. He, Y., et al.: Vista3d: Versatile imaging segmentation and annotation model for 3d computed tomography. *arXiv preprint [arXiv:2406.05285](https://arxiv.org/abs/2406.05285)* (2024)
13. Heinrich, M.P., Simpson, I.J., Papież, B.W., Brady, M., Schnabel, J.A.: Deformable image registration by combining uncertainty estimates from supervoxel belief propagation. *Med. Image Anal.* **27**, 57–71 (2016)
14. Hu, Q., et al.: Label-free liver tumor segmentation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7422–7432 (2023). <https://github.com/MrGiovanni/SyntheticTumors>
15. Isensee, F., Jaeger, P.F., Kohl, S.A., Petersen, J., Maier-Hein, K.H.: nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. *Nat. Methods* **18**(2), 203–211 (2021)
16. Isensee, F., et al.: nnu-net revisited: A call for rigorous validation in 3d medical image segmentation. *arXiv preprint [arXiv:2404.09556](https://arxiv.org/abs/2404.09556)* (2024)
17. Lai, Y., Chen, X., Wang, A., Yuille, A., Zhou, Z.: From pixel to cancer: cellular automata in computed tomography. *arXiv preprint [arXiv:2403.06459](https://arxiv.org/abs/2403.06459)* (2024). <https://github.com/MrGiovanni/Pixel2Cancer>
18. Li, J., et al.: Medshapenet—a large-scale dataset of 3d medical shapes for computer vision. *Biomed. Eng./Biomedizinische Technik* (0) (2024). <https://medshapenet.ikim.nrw>
19. Li, W., et al.: Scalemai: Accelerating the development of trusted datasets and ai models. *arXiv preprint [arXiv:2501.03410](https://arxiv.org/abs/2501.03410)* (2025). <https://github.com/MrGiovanni/ScaleMAI>
20. Li, W., et al.: Abdomenatlas: a large-scale, detailed-annotated, & multi-center dataset for efficient transfer learning and open algorithmic benchmarking. *Med. Image Anal.*, 103285 (2024). <https://github.com/MrGiovanni/AbdomenAtlas>
21. Li, W., Yuille, A., Zhou, Z.: How well do supervised models transfer to 3d image segmentation? In: International Conference on Learning Representations (2024). <https://github.com/MrGiovanni/SuPreM>
22. Li, X., et al.: Text-driven tumor synthesis. *arXiv preprint [arXiv:2412.18589](https://arxiv.org/abs/2412.18589)* (2024). <https://github.com/MrGiovanni/TextoMorph>
23. Litjens, G., et al.: A survey on deep learning in medical image analysis. *Med. Image Anal.* **42**, 60–88 (2017)
24. Liu, J., et al.: Clip-driven universal model for organ segmentation and tumor detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 21152–21164 (2023). <https://github.com/ljwztc/CLIP-Driven-Universal-Model>
25. Liu, J., et al.: Universal and extensible language-vision models for organ segmentation and tumor detection from abdominal computed tomography. *Med. Image Anal.* 103226 (2024). <https://github.com/ljwztc/CLIP-Driven-Universal-Model>
26. Ma, J., et al.: Abdomenct-1k: is abdominal organ segmentation a solved problem. *IEEE Trans. Pattern Anal. Mach. Intell.* (2021)

27. Mao, J., et al.: Medsegfactory: text-guided generation of medical image-mask pairs. arXiv preprint [arXiv:2504.06897](https://arxiv.org/abs/2504.06897) (2025). <https://github.com/jwmao1/MedSegFactory>
28. Park, S., et al.: Annotated normal ct data of the abdomen for deep learning: Challenges and strategies for implementation. *Diagn. Interv. Imaging* **101**(1), 35–44 (2020)
29. Qu, C., et al.: Abdomenatlas-8k: annotating 8,000 abdominal ct volumes for multi-organ segmentation in three weeks. In: *Conference on Neural Information Processing Systems*, vol. 21 (2023). <https://github.com/MrGiovanni/AbdomenAtlas>
30. Reinke, A., et al.: Common limitations of performance metrics in biomedical image analysis. *Nat. Commun.* **12**(1), 1–15 (2021)
31. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) *MICCAI 2015. LNCS*, vol. 9351, pp. 234–241. Springer, Cham (2015). [https://doi.org/10.1007/978-3-319-24574-4\\_28](https://doi.org/10.1007/978-3-319-24574-4_28)
32. Tajbakhsh, N., Jeyaseelan, L., Li, Q., Chiang, J.N., Wu, Z., Ding, X.: Embracing imperfect datasets: a review of deep learning solutions for medical image segmentation. *Med. Image Analy.*, 101693 (2020)
33. Wang, L., Li, D., Liu, H., Peng, J., Tian, L., Shan, Y.: Cross-dataset collaborative learning for semantic segmentation in autonomous driving. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, pp. 2487–2494 (2022)
34. Xia, Y., et al.: The felix project: deep networks to detect pancreatic neoplasms. *medRxiv* (2022)
35. Yang, Y., et al.: Medical world model: Generative simulation of tumor evolution for treatment planning. arXiv preprint [arXiv:2506.02327](https://arxiv.org/abs/2506.02327) (2025). <https://github.com/scott-yjyang/MeWM>
36. Zhang, T., Chen, X., Qu, C., Yuille, A., Zhou, Z.: Leveraging ai predicted and expert revised annotations in interactive segmentation: Continual tuning or full training? In: *IEEE International Symposium on Biomedical Imaging*. IEEE (2024). <https://github.com/MrGiovanni/ContinualLearning>
37. Zhang, Y., Li, X., Chen, H., Yuille, A.L., Liu, Y., Zhou, Z.: Continual learning for abdominal multi-organ and tumor segmentation. In: Greenspan, H., et al. (eds.) *MICCAI 2023. LNCS*, vol. 14221. Springer, Cham (2023). [https://doi.org/10.1007/978-3-031-43895-0\\_4](https://doi.org/10.1007/978-3-031-43895-0_4)
38. Zhou, Z., Gotway, M.B., Liang, J.: Interpreting medical images. In: *Intelligent Systems in Medicine and Health*, pp. 343–371. Springer (2022). [https://doi.org/10.1007/978-3-031-09108-7\\_12](https://doi.org/10.1007/978-3-031-09108-7_12)
39. Zhou, Z., Siddiquee, M.M.R., Tajbakhsh, N., Liang, J.: Unet++: redesigning skip connections to exploit multiscale features in image segmentation. *IEEE Trans. Med. Imaging* **39**(6), 1856–1867 (2019). <https://github.com/MrGiovanni/UNetPlusPlus>