



TorchIO

Q Search

[Getting started](#)[Data structures](#)[Patch-based pipelines](#)[Transforms](#)[Preprocessing](#)[Augmentation](#)[Others](#)[Medical image datasets](#)[Additional interfaces](#)[Examples gallery](#)[GitHub repository](#)[Paper](#)

## Others



CONTENTS

[Lambda](#)

### Lambda

```
class torchio.transforms.Lambda(function: Callable[[torch.Tensor], torch.Tensor],
                                types_to_apply: Optional[Sequence[str]] = None, **kwargs) [source]
```

Bases: [torchio.transforms.transform.Transform](#)

Applies a user-defined function as transform.

#### PARAMETERS

- **function** – Callable that receives and returns a 4D [torch.Tensor](#).
- **types\_to\_apply** – List of strings corresponding to the image types to which this transform should be applied. If `None`, the transform will be applied to all images in the subject.
- **\*\*kwargs** – See [Transform](#) for additional keyword arguments.

#### EXAMPLE

```
>>> import torchio as tio
>>> invert_intensity = tio.Lambda(lambda x: -x, types_to_apply=[tio.INTENSITY])
>>> invert_mask = tio.Lambda(lambda x: 1 - x, types_to_apply=[tio.LABEL])
>>> def double(x):
...     return 2 * x
>>> double_transform = tio.Lambda(double)
```

[Previous](#)  
[Augmentation](#)[Next](#)  
[Medical image datasets](#)

Copyright © 2022, Fernando Pérez-García | Created using Sphinx and [@pradyunsg's Furo theme](#). | [Show Source](#)

[v: latest](#)