1h_sed

November 28, 2022

[4]:
```
import numpy as np
import matplotlib.pyplot as plt
import hyperspy.api as hs

%matplotlib qt5
```

[2]:
```
s = hs.load("datasett/scanning_electron_diffraction_1.hspy")
s2 = hs.load("datasett/scanning_electron_diffraction_2.hspy")
```

[53]:
```
s.plot(norm="symlog")
```

[14]:
```
s2.plot(norm="symlog")
```

1 Pent

[17]:
```
from mpl_toolkits.axes_grid1.anchored_artists import AnchoredSizeBar
import matplotlib.font_manager as fm
import matplotlib.patheffects as patheffects
fontprops = fm.FontProperties(size=18)

scalebar_kwargs = {'size': 5, 'label': '5 $\mathrm{\mu}$m', 'loc': 4, 'frameon':
 ↪ False, 'color': 'white', 'size_vertical': 0.2, 'label_top': False,␣
 ↪'fontproperties': fontprops}
nice_effect = [patheffects.withStroke(linewidth=2, foreground='black',␣
 ↪capstyle="round")]

def add_scalebar(ax: plt.Axes):
    scalebar = AnchoredSizeBar(transform=ax.transData, **scalebar_kwargs)
    # Denne legger til et svart omriss rundt scalebar teksten, for å gjøre den␣
 ↪lettere å lese
    scalebar.txt_label._text.set_path_effects(nice_effect)
    ax.add_artist(scalebar)
```

[36]:
```
import matplotlib.colors as colors
norm = colors.SymLogNorm(1, vmin=1, vmax=s.inav[0].data.max())
```

[75]:

```
p1, p2, p3 = 10, 36, 55

fig, axdict = plt.subplot_mosaic(
    [
        ["left", "left", "top_right"],
        ["left", "left", "mid_right"],
        ["left", "left", "bot_right"],
    ]
)
for label, ax in axdict.items():
    if label == "left":
        ax.set_ylabel("intensity")
        ax.set_xlabel("position")
    else:
        ax.set_xticks([])
        ax.set_yticks([])
        ax.patch.set_linewidth(4)
        pass

axdict["top_right"].patch.set_edgecolor("red")
axdict["mid_right"].patch.set_edgecolor("blue")
axdict["bot_right"].patch.set_edgecolor("limegreen")

axdict["left"].plot(s.T.sum())
axdict["top_right"].imshow(s.inav[p1], norm=norm)
axdict["mid_right"].imshow(s.inav[p2], norm=norm)
axdict["bot_right"].imshow(s.inav[p3], norm=norm)

axdict["left"].axvline(p1, color="red")
axdict["left"].axvline(p2, color="blue")
axdict["left"].axvline(p3, color="limegreen")

axdict["top_right"].annotate("a", xy=(0.05, 0.85), xycoords="axes fraction",␣
 ↪color="white", fontsize=18)
axdict["mid_right"].annotate("b", xy=(0.05, 0.85), xycoords="axes fraction",␣
 ↪color="white", fontsize=18)
axdict["bot_right"].annotate("c", xy=(0.05, 0.85), xycoords="axes fraction",␣
 ↪color="white", fontsize=18)

axdict["left"].annotate("a", xy=(p1/len(s.inav[:]), 0.2), xycoords="axes␣
 ↪fraction", fontsize=18, color="red")
axdict["left"].annotate("b", xy=(p2/len(s.inav[:]), 0.2), xycoords="axes␣
 ↪fraction", fontsize=18, color="blue")
axdict["left"].annotate("c", xy=(p3/len(s.inav[:]), 0.2), xycoords="axes␣
 ↪fraction", fontsize=18, color="limegreen")

fig.tight_layout(pad=0.5)
fig.savefig("sed_1", dpi=300, pad_inches=0, bbox_inches="tight")
```

[76]:

```
p1, p2, p3 = 10, 36, 55

fig, axdict = plt.subplot_mosaic(
    [
        ["left", "left", "top_right"],
        ["left", "left", "mid_right"],
        ["left", "left", "bot_right"],
    ]
)
for label, ax in axdict.items():
    if label == "left":
        ax.set_ylabel("intensity")
        ax.set_xlabel("position")
    else:
        ax.set_xticks([])
        ax.set_yticks([])
        ax.patch.set_linewidth(4)
        pass

axdict["top_right"].patch.set_edgecolor("red")
axdict["mid_right"].patch.set_edgecolor("blue")
axdict["bot_right"].patch.set_edgecolor("limegreen")

axdict["left"].plot(s2.T.sum())
axdict["top_right"].imshow(s2.inav[p1], norm=norm)
axdict["mid_right"].imshow(s2.inav[p2], norm=norm)
axdict["bot_right"].imshow(s2.inav[p3], norm=norm)

axdict["left"].axvline(p1, color="red")
axdict["left"].axvline(p2, color="blue")
axdict["left"].axvline(p3, color="limegreen")

axdict["top_right"].annotate("a", xy=(0.05, 0.85), xycoords="axes fraction",␣
  ↪color="white", fontsize=18)
axdict["mid_right"].annotate("b", xy=(0.05, 0.85), xycoords="axes fraction",␣
  ↪color="white", fontsize=18)
axdict["bot_right"].annotate("c", xy=(0.05, 0.85), xycoords="axes fraction",␣
  ↪color="white", fontsize=18)

axdict["left"].annotate("a", xy=(p1/len(s2.inav[:]), 0.2), xycoords="axes␣
  ↪fraction", fontsize=18, color="red")
axdict["left"].annotate("b", xy=(p2/len(s2.inav[:]), 0.2), xycoords="axes␣
  ↪fraction", fontsize=18, color="blue")
axdict["left"].annotate("c", xy=(p3/len(s2.inav[:]), 0.2), xycoords="axes␣
  ↪fraction", fontsize=18, color="limegreen")
```

```python
fig.tight_layout(pad=0.5)
fig.savefig("sed_2", dpi=300, pad_inches=0, bbox_inches="tight")
```