Matter, Volume *1*

Supplemental Information

Inverse Design of Solid-State Materials via a Continuous Representation

Juhwan Noh, Jaehoon Kim, Helge S. Stein, Benjamin Sanchez-Lengeling, John M. Gregoire, Alan Aspuru-Guzik, and Yousung Jung

S1. Machine learning model details

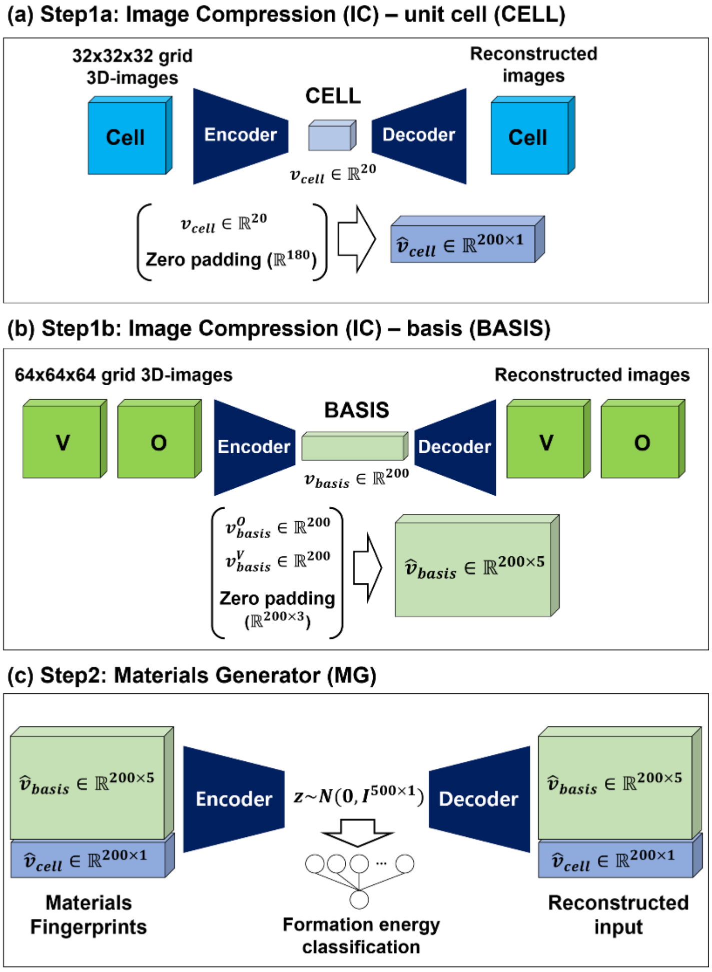

(a) Step1a: Image Compression (IC) – unit cell (CELL)
32x32x32 grid 3D-images
Cell
Encoder
CELL
Decoder
Reconstructed images
Cell
$v_{cell} \in \mathbb{R}^{20}$
$v_{cell} \in \mathbb{R}^{20}$
Zero padding ($\mathbb{R}^{180}$)
$\hat{v}_{cell} \in \mathbb{R}^{200\times1}$
(b) Step1b: Image Compression (IC) – basis (BASIS)
64x64x64 grid 3D-images
V
O
Encoder
BASIS
Decoder
Reconstructed images
V
O
$v_{basis} \in \mathbb{R}^{200}$
$v_{basis}^{O} \in \mathbb{R}^{200}$
$v_{basis}^{V} \in \mathbb{R}^{200}$
Zero padding ($\mathbb{R}^{200\times3}$)
$\hat{v}_{basis} \in \mathbb{R}^{200\times5}$
(c) Step2: Materials Generator (MG)
$\hat{v}_{basis} \in \mathbb{R}^{200\times5}$
$\hat{v}_{cell} \in \mathbb{R}^{200\times1}$
Materials Fingerprints
Encoder
$z \sim N(0, I^{500\times1})$
Decoder
Formation energy classification
$\hat{v}_{basis} \in \mathbb{R}^{200\times5}$
$\hat{v}_{cell} \in \mathbb{R}^{200\times1}$
Reconstructed input

Figure S1. The proposed hierarchical two-step Image-based Materials Generator (iMatGen); (a) image compression for unit cell and (b) basis, and (c) materials generator (MG) using a variational autoencoder (VAE) model.

The structure of the 3D convolutional autoencoder was inspired from the 3D GAN introduced by Wu and coauthors[1], and details are described in S1.1 and S1.2 where all machine learning models were developed using *Tensorflow*[2], an open-source machine learning framework using *Python.*

S1.1 Cell convolutional autoencoder (CELL)

For the CELL (Figure S1a), the encoder consists of four 3D convolutional layers with kernel sizes {4,4,4,4}, strides {2S,2S,2S,1V} and number of channels {64,64,64,20}. For stride, S and V mean padding option SAME and VALID used in *Tensorflow*, respectively. We used a leaky-ReLU activation function with a parameter of 0.2 between convolutional layers and hyperbolic tangent function at the end layer. The input is a 32×32×32 tensor, and the output (or the latent vector, v_{cell}) is a 20-dimensional vector the value of which is in [-1, 1].

The decoder has a mirrored structure to that of encoder where the kernel sizes are {4,4,4,4}, the strides are {1V,2S,2S,2S} and the number of channels are {64,64,64,1}. We used the leaky-ReLU activation function with a parameter of 0.2 between convolutional layers and sigmoid function at the end. The decoder takes the input a 20-dimensional vector (v_{cell}), and the output a 32×32×32 tensor the value of which is in [0, 1].

For training of CELL, we adopted mean-squared-error loss function (or L2-loss function).

S1.2 Basis convolutional autoencoder (BASIS)

For the BASIS (Figure S1b), the encoder consists of five 3D convolutional layers with kernel sizes {4,4,4,4,4}, strides {2S,2S,2S,2S,1V} and number of channels {64,64,64,64,200}. We used a leaky-ReLU activation function with parameter of 0.2 between convolutional layers, and hyperbolic tangent function at the end. The input is a 64×64×64 tensor, and the output (or the latent vector, v_{basis}) is a 500-dimensional vector the value of which is in [-1, 1].

The decoder has a mirrored structure to that of encoder where kernel sizes {4,4,4,4,4}, strides {1V,2S,2S,2S,2S} and number of channels {64,64,64,64,1}. We used a leaky-ReLU activation function with the amount of leakage is 0.2 between convolutional layers, and sigmoid function at the end. The decoder takes the input a 200-dimensional vector (v_{basis}), and the output is a 64×64×64 tensor the value of which is in [0, 1].

For training of BASIS, we adopted mean-squared-error loss function (or L2-loss function).

S1.3 Materials Generator (MG) – Variational AutoEncoder (VAE)

The architecture of VAE (Figure S1c) is following. The encoder consists of four 2D convolutional layers with kernel sizes {4,4,4,1}, strides {2S,2S,2S,1S} and number of channels {100,100,100,50}, and we used leaky-ReLU activation function with parameter 0.2 between convolutional layers. The input is a 1×200×6 tensor, and the output is a 1250-dimensional vector. The output of the convolutional layers are transformed to a 500-dimensional vector (z) by fully connected layers. Between the last convolutional layer and fully connected layers, a leaky ReLU with parameter 0.2 was used.

The decoder takes a 500-dimensional vector (z) as input which is transformed to a 1250-dimensional vector by a fully connected layer with leaky ReLU with parameter 0.2. Next, four 2D convolutional layers were followed with kernel sizes {1,4,4,4}, strides {1S,2S,2S,2S} and number of channels {100,100,100,6}. A leaky ReLU with parameter of 0.2 was used between convolutional layers, and hyperbolic tangent activation function at the end.

For the formation energy classification task from the latent space, we used a single hidden layer neural network where the input is a 500-dimensional vector (z) and the number of node in hidden layer is 500. A Hyperbolic tangent function was used between layers and sigmoid function was used at the end layer.

For training of VAE, we adopted following loss function, $\mathcal{L}_{VAE}$:

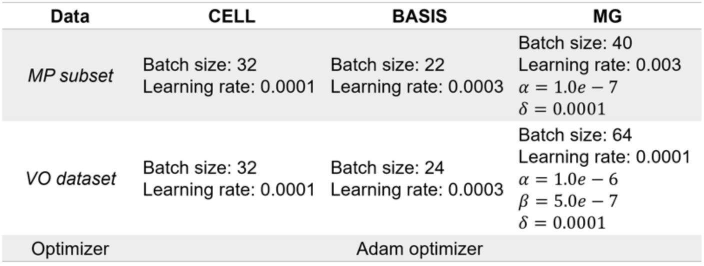
$$\mathcal{L}_{VAE} = \mathcal{L}_{recon.} + \alpha\mathcal{L}_{KL} + \beta\mathcal{L}_{class.}, \;\; z \sim \delta * N(0, I)$$

where $\mathcal{L}_{recon.}$ is the reconstruction loss term (L2-loss) and $\mathcal{L}_{KL}$ is the KL divergence term to regularize the latent space (z) to the normal distribution. Furthermore, $\mathcal{L}_{class.}$ represents classification loss where we used the cross entropy binary classification loss with the weighting parameter β to control learning speed, similar to α for the $\mathcal{L}_{KL}$.

S1.4 Training notes

For all training procedures, we randomly sampled 90% of the total data as a training set. Before training iMatGen with the *VO dataset*, we trained iMatGen using the *MP subset* consisting of 28,242 crystal structures taken from the MP database (up to 5 atom types in the unit cell) satisfying two constraints as mentioned in the main text to obtain pre-trained model parameters for **IC** and **MG** (without formation energy classification task). Training of iMatGen with *VO dataset* was done by using the latter pre-trained model parameters, and as mentioned in the main text, formation energy classification task was also additionally included. All the used hyper-parameters including batch size, learning rate and optimizer are listed in Table S1.

Table S1. Lists of the all hyper-parameters used in this work



Data	CELL	BASIS	MG
MP subset	Batch size: 32 Learning rate: 0.0001	Batch size: 22 Learning rate: 0.0003	Batch size: 40 Learning rate: 0.003 $\alpha = 1.0e-7$ $\delta = 0.0001$
VO dataset	Batch size: 32 Learning rate: 0.0001	Batch size: 24 Learning rate: 0.0003	Batch size: 64 Learning rate: 0.0001 $\alpha = 1.0e-6$ $\beta = 5.0e-7$ $\delta = 0.0001$
Optimizer	Adam optimizer		

S1.5 Latent space of the trained MG

Since the normal distribution is assumed as prior distribution of the latent space, we checked that the trained latent space is shaped appropriately. As shown in Figure S2, the mean and standard deviation of the latent space taken from the trained MG are well matched to the normal distribution.

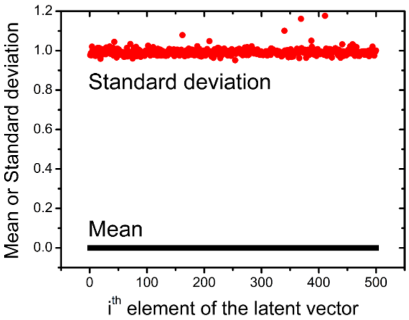


Figure S2. Mean and standard deviation distribution of the 500-latent vector elements after training the Materials Generator (MG)

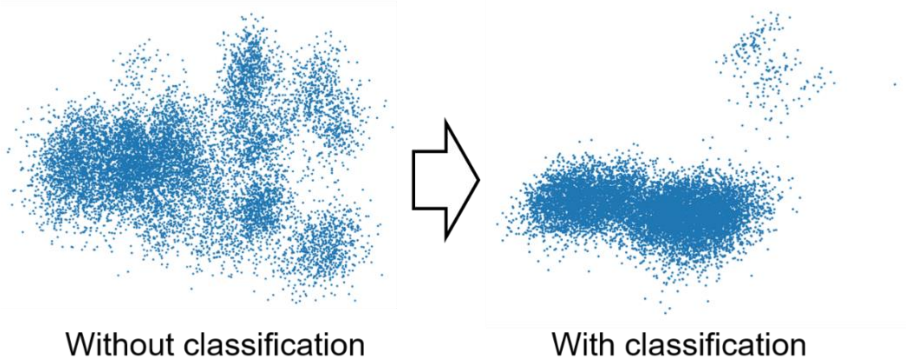


Figure S3. Effect of including stability classification task from the latent space of the Materials Generator (MG)

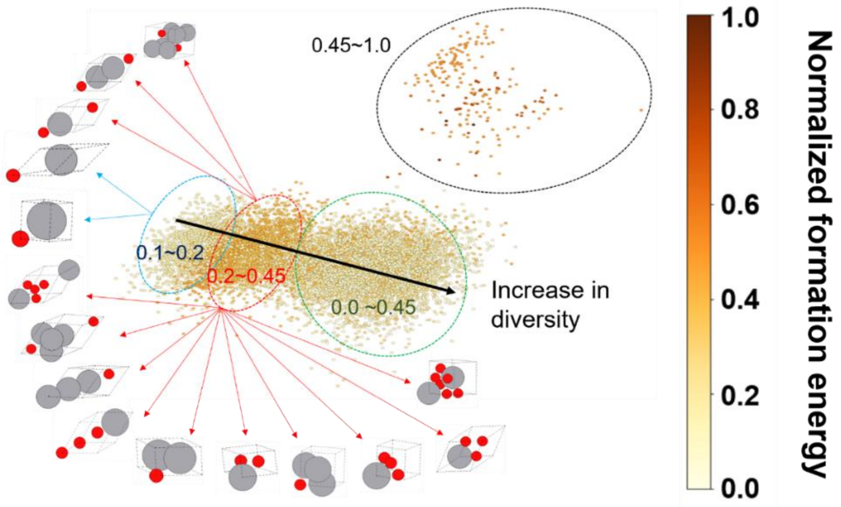


Figure S4. Statistical analysis of the trained latent space with respect to the stability and structural diversity

S2. DFT calculation details

S2.1 DFT calculations for *VO dataset*

We used the PBE functional[3] and the PAW[4]-PBE pseudopotentials for the V and O atoms as implemented in the program package VASP[5]. Structures (both atomic positions and cell parameters) were fully relaxed using the conjugate gradient descent method with convergence criteria of 1.0e-5 for energy and 0.05 $\mathrm{eV/Å}$ for force. For computational efficiency, we used relatively sparse reciprocal space meshes (0.5 $\mathrm{Å^{-1}}$ as the grid spacing) and the cut-off energy for the plane-wave expansions of 500 eV since the main goal of constructing the VO dataset was collecting diverse atomic configurations. The formation energy (or formation enthalpy, E_f) was calculated using $\mathrm{E_{V_xO_y} - (xE_V - yE_O)/(x+y)}$ (in eV/atom).

S2.2 DFT calculations for the generated materials

For all generated structures (~20,000 materials), we performed spin polarized GGA+U calculations, with the same U parameter for V used in the Materials Projects database[6]. We relaxed both atomic positions and cell parameters using conjugate gradient descent method with convergence criteria of 1.0e-5 for energy and 0.05 $\mathrm{eV/Å}$ for force with 500 eV cut-off energy. To compare the phase stability among the generated structures for all generated materials we first used sparse reciprocal lattice grid with a grid spacing of 0.5 $\mathrm{Å^{-1}}$ (See Figure 4 and Figure 6). Then, for a smaller set of materials that satisfy $E_{\mathrm{hull}} \le 0.1\ \mathrm{eV/atom}$ for *Slerp* and $E_{\mathrm{hull}} \le 0.2\ \mathrm{eV/atom}$ for *Random*), we refined the formation energy calculations using a denser reciprocal lattice grid with grid spacing of 0.25 $\mathrm{Å^{-1}}$ (Figure 7).

S2.3 Phase Diagram using GGA/GGA+*U*-mixed approach

As mentioned in the main text, we followed the approach proposed by Anubhav Jain et al.[7] which used GGA/GGA+*U*-mixed approach for the formation energy calculation especially for transition metal oxides. In the latter approach, the energy of all GGA+*U* calculations is corrected by equation S1.

$$E_{V,O\ \mathrm{compound}}^{\mathrm{GGA+U\ renorm.}} = E_{V,O\ \mathrm{compound}}^{\mathrm{GGA+U}} - n_V \Delta E_V \qquad \text{(S1)}$$

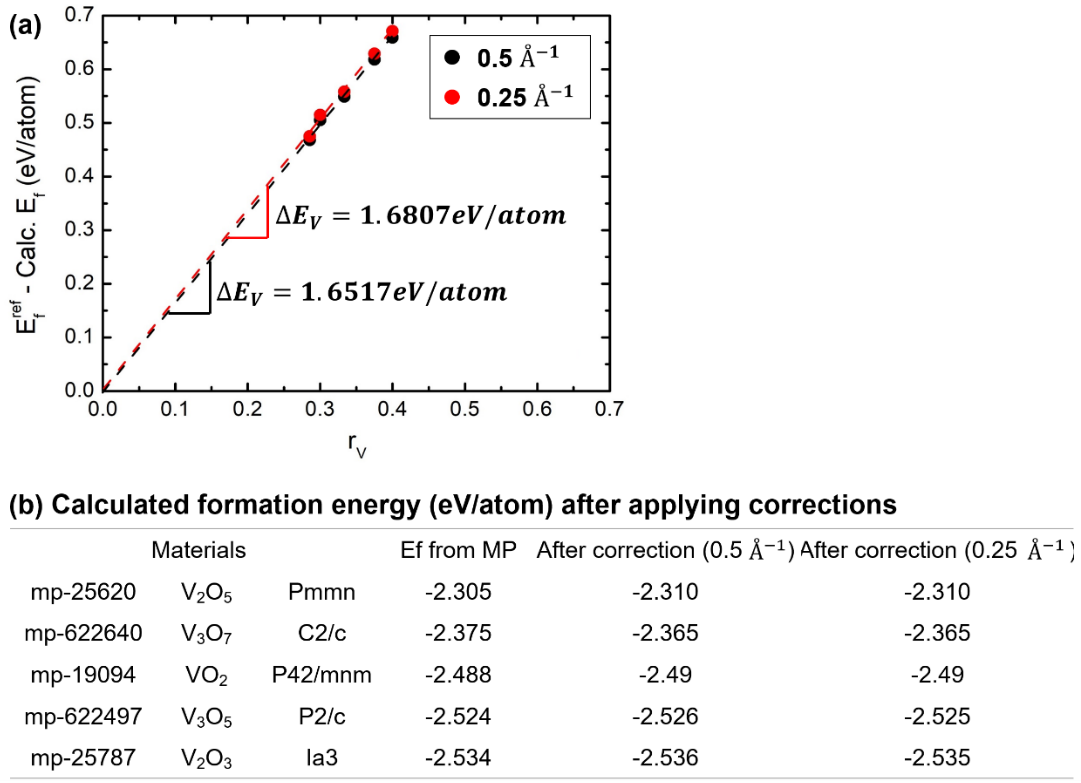
In equation (S1), $E_{V,O\ \mathrm{compound}}^{\mathrm{GGA+U}}$ is the GGA+U energy for V, O compound and n_V is the number of V atoms in the compound. Here, our target is to obtain a correction term ΔE_V for V. In addition, we included an O_2 energy correction term (equation S2) taken from the MP database for the $E_{V,O\ \mathrm{compound}}^{\mathrm{GGA+U}}$ energy.

$$E(O_2)_{corr.} = -1.4046 eV/atom \qquad \text{(S2)}$$

After a simple rearrangement using equation (S1), the correction energy for V atom is then obtained using equation (S3).

$$\Delta E_V = (E_f^{\text{ref.}} - \text{Calc.}\, E_f)/r_V \qquad \text{(S3)}$$

In equation (S3), $E_f^{\text{ref.}}$ is the formation energy for the 5-reference structures (V_2O_5, V_3O_7, VO_2, V_3O_5 and V_2O_3) taken from the MP database all of which are at the ground state for the corresponding composition, $\text{Calc.}\, E_f$ is the formation energy calculated from this work, and r_V is the fraction of V atom in a compound. Therefore, ΔE_V can be interpreted as the slope of the $(E_f^{\text{ref.}} - \text{Calc.}\, E_f)$ vs. r_V plot as shown in Figure S5a. We calculated ΔE_V for two cases of reciprocal lattice grid space (0.5 $Å^{-1}$ and 0.25 $Å^{-1}$). The value of ΔE_V for each case is shown in Figure S5a, and the adjusted formation energy values are described in Figure S5b.



(b) Calculated formation energy (eV/atom) after applying corrections

Materials			Ef from MP	After correction (0.5 $Å^{-1}$)	After correction (0.25 $Å^{-1}$)
mp-25620	V_2O_5	Pmmn	-2.305	-2.310	-2.310
mp-622640	V_3O_7	C2/c	-2.375	-2.365	-2.365
mp-19094	VO_2	P42/mnm	-2.488	-2.49	-2.49
mp-622497	V_3O_5	P2/c	-2.524	-2.526	-2.525
mp-25787	V_2O_3	Ia3	-2.534	-2.536	-2.535

Figure S5. Results of GGA/GGA+U mixing scheme applied in this work: (a) Deriving correction energy for V (ΔE_V) using the GGA/GGA+U-mixed scheme for the 5-V_xO_y materials at the convex hull, and (b) results after applying the latter correction energy terms for the 5-V_xO_y materials at the convex hull (in eV/atom).

As shown in the Figure S5b, we find that all the formation energies after correction are well matched to the reference values. For further validation, we next applied the same correction schemes to calculate phase stability (formation energy and energy above the convex hull) of all 112 vanadium oxide structures existing in MP database. We find a good agreement between the present results and those from the MP database as shown in Figure S6.

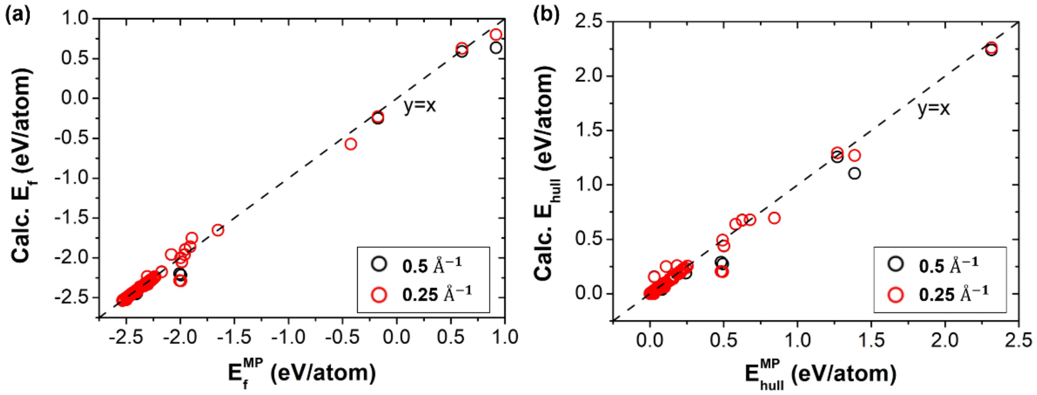


Figure S6. Comparison of the calculated phase stability values for 112-VxOy materials existing in MP database: (a) formation energy (E_f) and (b) energy above the convex hull (E_{hull}).

S3. Materials Generation

S3.1 Latent space sampling

In addition to Gaussian random sampling of 10,000 vectors from the latent space, we additionally sampled the latent vector by interpolating between the known V_xO_y structures in MP database with the assumption that the interpolated points between two existing structures can generate more reasonable structures. Here, we applied the **s**pherical **l**inear int**erp**olation (***slerp***, (equation S4)):

$$z_m = \frac{\sin(t\cdot\Omega)}{\sin\Omega} z_{start} + \frac{\sin(1-t)\cdot\Omega}{\sin\Omega} z_{end},\ t = m/N \qquad \text{(S4)}$$

where z_m is the interpolated vector, Ω is the angle between two latent vectors (z_{start} and z_{end}). t is the interpolation grid defined as m/N, where the maximum number of divisions (N) is 20 and m is positive integer in [1,19] ($m = 0$ is corresponding to z_{start} and $m = 20$ is corresponding to z_{end}).

S3.2 Inverse transformation of the input representation

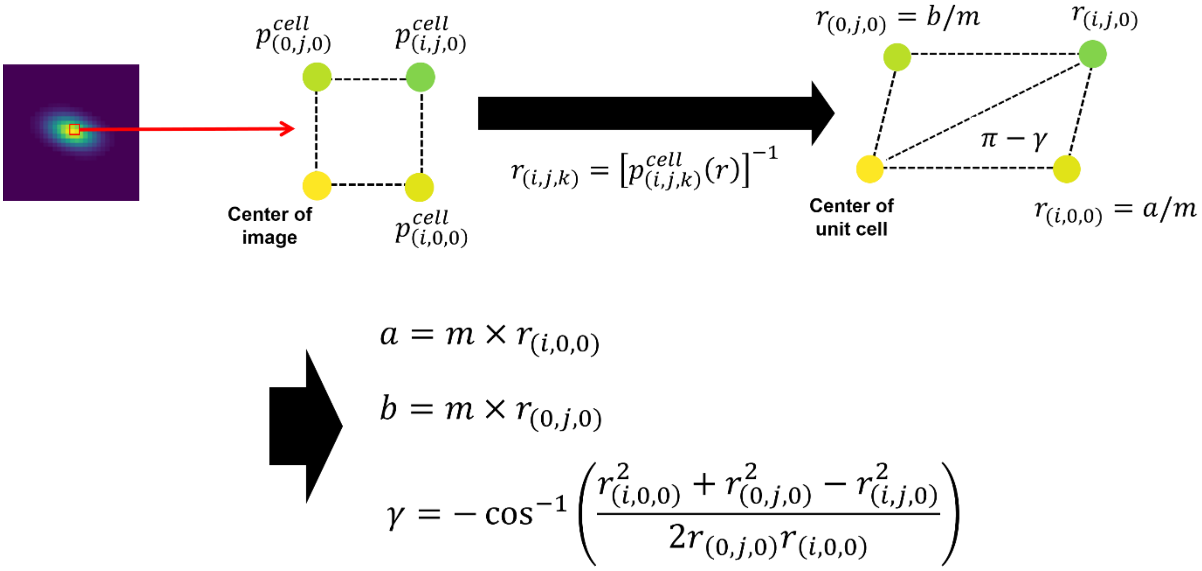

$p^{cell}_{(0,j,0)}$
$p^{cell}_{(i,j,0)}$
Center of image
$p^{cell}_{(i,0,0)}$
$r_{(i,j,k)} = [p^{cell}_{(i,j,k)}(r)]^{-1}$
$r_{(0,j,0)} = b/m$
$r_{(i,j,0)}$
$\pi - \gamma$
Center of unit cell
$r_{(i,0,0)} = a/m$
$a = m \times r_{(i,0,0)}$
$b = m \times r_{(0,j,0)}$
$\gamma = -\cos^{-1}\left(\frac{r^2_{(i,0,0)} + r^2_{(0,j,0)} - r^2_{(i,j,0)}}{2r_{(0,j,0)}r_{(i,0,0)}}\right)$

Figure S7. Description of unit cell reconstruction from input representation: Herein, we show reconstruction of ab-plane as an example.

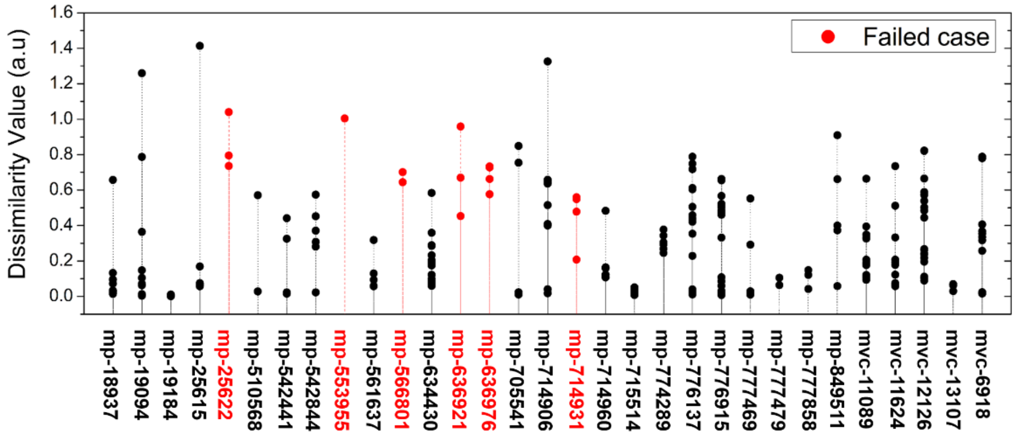


Figure S8. Dissimilarity value for the reconstructed materials with respect to the reference materials as shown in Fig 4 of main text.

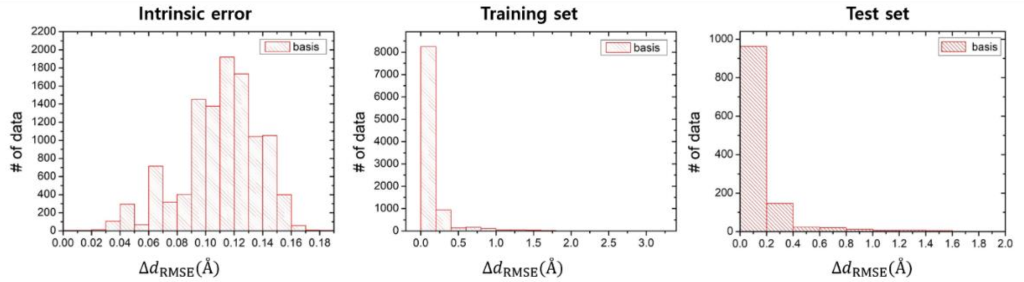


Figure S9. Error histograms for the reconstruction scheme applied to the basis images.

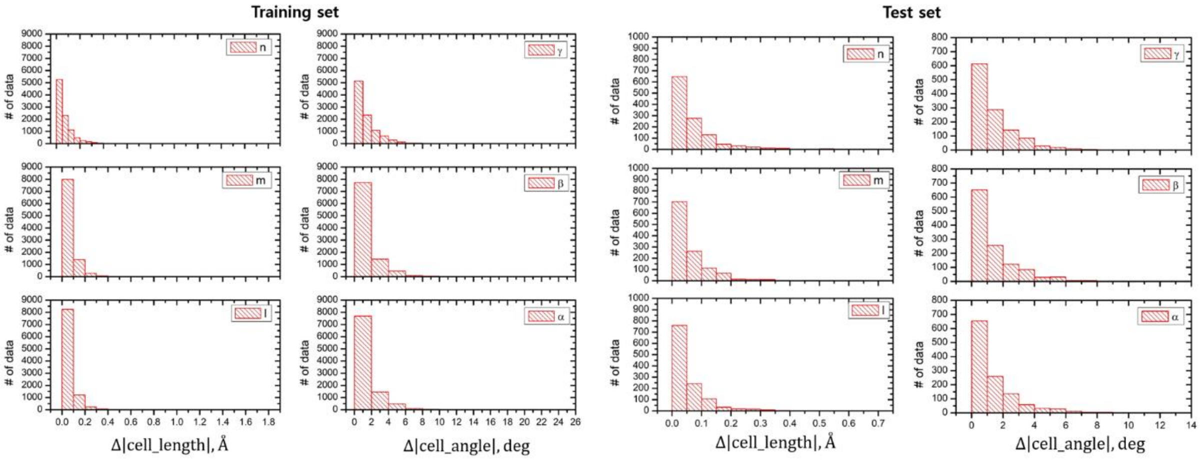


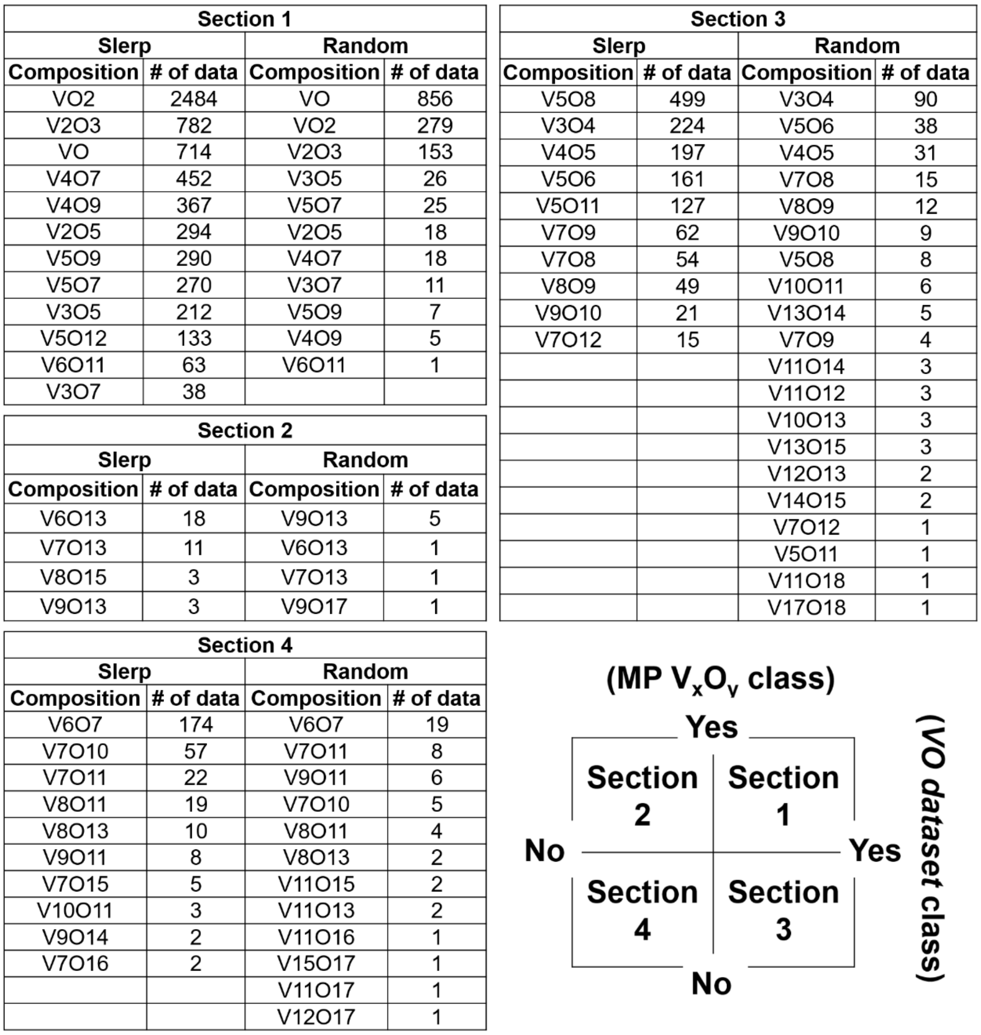
Figure S10. Error histograms for the inverse transform scheme applied to the cell images

S3.3 Post-process for the generated images

In post-processing of the generated structures, we first removed the unfavorable position overlap within a single image under periodic boundary conditions (PBC). In other words, we imposed the PBC of each generated image in a post-hoc manner by merging the V atoms within the covalent radius (1.6 Å) as a single V atom at the unit cell boundary, and likewise for the oxygen images using the 1.3 Å criterion. We used the same criteria to handle two atoms of the same atom type in close proximity inside the unit cell. For structures in which V and O are in close proximity, we used 1.2 Å criterion to remove oxygen atom in a way to meet the valid oxidation state of V.

S3.4 Materials generation from the iMatGen – numerical statistics

Figure S11 shows the distribution of all generated V_xO_y compositions within valid OS_V range as shown in Figure 5 of the main text, and 4-different sections are also described in the inset of Figure S11.



Section 1			
Slerp		Random	
Composition	# of data	Composition	# of data
VO2	2484	VO	856
V2O3	782	VO2	279
VO	714	V2O3	153
V4O7	452	V3O5	26
V4O9	367	V5O7	25
V2O5	294	V2O5	18
V5O9	290	V4O7	18
V5O7	270	V3O7	11
V3O5	212	V5O9	7
V5O12	133	V4O9	5
V6O11	63	V6O11	1
V3O7	38		

Section 2			
Slerp		Random	
Composition	# of data	Composition	# of data
V6O13	18	V9O13	5
V7O13	11	V6O13	1
V8O15	3	V7O13	1
V9O13	3	V9O17	1

Section 4			
Slerp		Random	
Composition	# of data	Composition	# of data
V6O7	174	V6O7	19
V7O10	57	V7O11	8
V7O11	22	V9O11	6
V8O11	19	V7O10	5
V8O13	10	V8O11	4
V9O11	8	V8O13	2
V7O15	5	V11O15	2
V10O11	3	V11O13	2
V9O14	2	V11O16	1
V7O16	2	V15O17	1
		V11O17	1
		V12O17	1

Section 3			
Slerp		Random	
Composition	# of data	Composition	# of data
V5O8	499	V3O4	90
V3O4	224	V5O6	38
V4O5	197	V4O5	31
V5O6	161	V7O8	15
V5O11	127	V8O9	12
V7O9	62	V9O10	9
V7O8	54	V5O8	8
V8O9	49	V10O11	6
V9O10	21	V13O14	5
V7O12	15	V7O9	4
		V11O14	3
		V11O12	3
		V10O13	3
		V13O15	3
		V12O13	2
		V14O15	2
		V7O12	1
		V5O11	1
		V11O18	1
		V17O18	1

Figure S11. All numerical results for the generated materials within the valid OS_V range. Each section is depicted in the inset coordination figure.

References

1 Wu, J., Zhang, C., Xue, T., Freeman, B., and Tenenbaum, J. (2016). Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. Paper presented at: Advances in Neural Information Processing Systems.

2 Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., and Isard, M. (2016). Tensorflow: a system for large-scale machine learning. Paper presented at: OSDI.

3 Perdew, J.P., Burke, K., and Ernzerhof, M. (1996). Generalized gradient approximation made simple. Physical review letters *77*, 3865.

4 Blöchl, P.E. (1994). Projector augmented-wave method. Physical Review B *50*, 17953.

5 Kresse, G., and Furthmüller, J. (1996). Software VASP, vienna (1999). Phys Rev B *54*, 169.

6 Jain, A., Ong, S.P., Hautier, G., Chen, W., Richards, W.D., Dacek, S., Cholia, S., Gunter, D., Skinner, D., and Ceder, G. (2013). Commentary: The Materials Project: A materials genome approach to accelerating materials innovation. Apl Materials *1*, 011002.

7 Jain, A., Hautier, G., Ong, S.P., Moore, C.J., Fischer, C.C., Persson, K.A., and Ceder, G. (2011). Formation enthalpies by mixing GGA and GGA+Ucalculations. Physical Review B *84*.